JC13 Rec'd PCT/PTO 3 0 MAR 2001

Form PTO-1390 (REV 10-2000) U S DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE

TRANSMITTAL LETTER TO THE UNITED STATES DESIGNATED/ELECTED OFFICE (DO/EO/US) CONCERNING A FILING UNDER 35 U.S.C. 371

ATTORNEY'S DOCKET NUMBER	
1797.014PC02	
U.S. APPLICATION NO. (IF KNOWN, SEE 37 C F.R. § 1 5)	
To be assigned 09/806398	

INTERNATIONAL APPLICATION NO	INTERNATIONAL FILING DATE	PRIORITY DATE CLAIMED
PCT/US99/22710	01 October 1999	01 October 1998

TITLE OF INVENTION

Distributed Shared Key Generation and Management Using Fractional Keys

APPLICANT(S) FOR DO/EO/US

University of Maryland *et al.*

Applicant herewith submits to the United States Designated/Elected Office (DO/EO/US) the following items and other information:

1. ☒ This is a **FIRST** submission of items concerning a filing under 35 U.S.C. 371.
2. ☐ This is a **SECOND** or **SUBSEQUENT** submission of items concerning a filing under 35 U.S.C. 371.
3. ☒ This is an express request to begin national examination procedures (35 U.S.C. 371(f)).
4. ☒ The US has been elected by the expiration of 19 months from the priority date (PCT Article 31).
5. ☒ A copy of the International Application as filed (35 U.S.C. 371(c)(2))
   a. ☐ is attached hereto (required only if not communicated by the International Bureau).
   b. ☐ has been communicated by the International Bureau.
   c. ☒ is not required, as the application was filed in the United States Receiving Office (RO/US).
6. ☐ An English language translation of the International Application as filed (35 U.S.C. 371(c)(2)).
7. ☒ Amendments to the claims of the International application under PCT Article 19 (35 U.S.C. 371(c)(3))
   a. ☐ are attached hereto (required only if not communicated by the International Bureau).
   b. ☐ have been communicated by the International Bureau.
   c. ☐ have not been made; however, the time limit for making such amendments has NOT expired.
   d. ☒ have not been made and will not be made.
8. ☐ An English language translation of the amendments to the claims under PCT Article 19 (35 U.S.C. 372(c)(3)).
9. ☐ An oath or declaration of the inventor(s) (35 U.S.C. 371(c)(4)).
10. ☐ An English language translation of the annexes to the International Preliminary Examination Report under PCT Article 36 (35 U.S.C. 371(c)(5)).

Items 11. to 16. below concern other document(s) or information included:

11. ☐ An Information Disclosure Statement under 37 C.F.R. 1.97 and 1.98.

12. ☐ An assignment document for recording. A separate cover sheet in compliance with 37 C.F.R. 3.28 and 3.31 is included.

13. ☐ A FIRST preliminary amendment.
    ☐ A SECOND or SUBSEQUENT preliminary amendment.

14. ☐ A substitute specification.

15. ☐ A change of power of attorney and/or address letter.

16. ☒ Other items or information:

    a. Authorization to Treat a Reply as Incorporating An Extension of Time Under 37 C.F.R. § 1.136(a)(3)
    b. Power of Attorney from Assignee
    c. Certificate Under 37 C.F.R. § 3.73(b)

U.S. APPLICATION NO. (if known, see 37 C F R 1 50)	INTERNATIONAL APPLICATION NO	ATTORNEY'S DOCKET NUMBER
09/806398	PCT/US99/22710	1797.014PC02

17. ☒ The following fees are submitted:	CALCULATIONS	PTO USE ONLY
Basic National Fee (37 CFR 1.492(a)(1)-(5)): Neither international preliminary examination fee (37 CFR 1.482) nor international search fee (37 CFR 1.445(a)(2)) paid to USPTO and International Search Report not prepared by the EPO or JPO **$1000.00**		
International preliminary examination fee (37 CFR 1.482) not paid to USPTO but International Search Report prepared by the EPO or JPO **$860.00**		
International preliminary examination fee (37 CFR 1.482) not paid to USPTO but international search fee (37 CFR 1.445(a)(2)) paid to USPTO **$710.00**		
International preliminary examination fee paid to USPTO (37 CFR 1.482) but all claims did not satisfy provisions of PCT Article 33(1)-(4) **$690.00**		
International preliminary examination fee paid to USPTO (37 CFR 1.482) and all claims satisfied provisions of PCT Article 33(2)-(4) ..... **$ 100.00**		
ENTER APPROPRIATE BASIC FEE AMOUNT =	$690.00	
Surcharge of **$130.00** for furnishing the oath or declaration later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492(e)).	$	

Claims	Number Filed		Number Extra	Rate		
Total Claims	16- 20	=	0	X **$18.00**	$ 00.00	
Independent Claims	3- 3	=	0	X **$80.00**	$00.00	
Multiple dependent claim(s) (if applicable)				+ **$270.00**	$ 00.00	
TOTAL OF ABOVE CALCULATIONS =					$690.00	
☐ Applicant claims small entity status. See 37 CFR 1.27. The fees indicated above are reduced by 1/2.					$	
SUBTOTAL =					$690.00	
Processing fee of $130.00 for furnishing the English translation later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492(f)). +					$	
TOTAL NATIONAL FEE =					$690.00	
Fee for recording the enclosed assignment (37 CFR 1.21(h)). The assignment must be accompanied by an appropriate cover sheet (37 CFR 3.28, 3.31). **$40.00** per property +					$	
TOTAL FEES ENCLOSED =					$690.00	
					Amount to be refunded:	$
					charged:	$

a. ☒ A check in the amount of $690.00 to cover the above fees is enclosed.

b. ☐ Please charge my Deposit Account No. _____ in the amount of $ _____ to cover the above fees. A duplicate copy of this sheet is enclosed.

c. ☒ The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. 19-0036. A duplicate copy of this sheet is enclosed.

NOTE: Where an appropriate time limit Under 37 CFR 1.494 or 1.495 has not been met, a petition to revive (37 CFR 1.137(a) or (b)) must be filed and granted to restore the application to pending status.

SEND ALL CORRESPONDENCE TO:
STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 New York Avenue, NW, Suite 600
Washington, D.C. 20005-3934

Edward W. Yee 3/30/01
SIGNATURE

Edward W. Yee
NAME

47,294
REGISTRATION NUMBER

Rec'd PCT/PTO 23 JUL 2001

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE #4/a

In re application of:	
Poovendran *et al.*	Art Unit: *To be assigned*
Appl. No. 09/806,398	Examiner: *To be assigned*
Filed: March 30, 2001	Atty. Docket: 1797.0140002
For: **Distributed Shared Key Generation and Management Using Fractional Keys**	

Preliminary Amendment Under 37 C.F.R. § 1.115

09806398 033001

Commissioner for Patents
Washington, D.C. 20231

Sir:

Submitted herein is a Preliminary Amendment Under 37 C.F.R. § 1.115.

This Amendment is provided in the following format:

(A) A clean version of each replacement paragraph/section/claim along with clear instructions for entry; and

(B) Starting on a separate page, appropriate remarks and arguments. 37 C.F.R. § 1.111 and MPEP 714.

It is not believed that extensions of time or fees for net addition of claims are required beyond those that may otherwise be provided for in documents accompanying this paper. However, if additional extensions of time are necessary to prevent abandonment of this application, then such extensions of time are hereby petitioned under 37 C.F.R. § 1.136(a), and any fees required therefor (including fees for net addition of claims) are hereby authorized to be charged to our Deposit Account No. 19-0036.

Amendments

In the Specification:

On page 1, in the section "Background of the Invention", in the subsection "Statement as to Rights to Inventions Made Under Federally-Sponsored Research and Development", please substitute the sole pending paragraph with the following paragraph:

This invention was made with U.S. Government support under Contract Number CG9813, awarded by the National Security Agency, and Contract Number DAAL019620002, awarded by the Army Research Laboratory. The U.S. Government has certain rights in this invention.

Remarks

The amendment above adds no new matter.

The amendment corrects a formal matter without changing the scope of the claims.

Accordingly, Applicants respectfully request that this Amendment be entered.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.

Edward W. Yee
Attorney for Applicants
Registration No. 47,294

Date: 7/23/01

1100 New York Avenue, N.W.
Suite 600
Washington, D.C. 20005-3934
(202) 371-2600

C:\TEMP\PreAm
SKGF Rev 3/16/01

Distributed Shared Key Generation and Management Using Fractional Keys

FOR EXAMPLE ONLY

Field of the Invention

The invention described herein pertains to communications, and more particularly to information security.

Related Art

Cryptographic key generation and management is an important problem in multicast and group communications (R. Canetti and Pinkas, B., "A taxonomy of multicast security issues," in *Internet-Draft* (1998); Harney, H. and Muckenhirn, C., "GKMP Architecture," *RFC 2093* (1997); Harney, H. and Muckenhirn, C., "GKMP Architecture," *RFC 2094* (1997); Ballardie, A., "Scalable Multicast Key Distribution," *RFC 1949* (1996); Poovendran, R., *et al.*, "A Scalable Extension of Group Key Management Protocol," *Proc. 2nd Ann.*

ATIRP Conf., Maryland, pp. 187-191 (1998), incorporated herein by reference). In many instances, it is desirable to generate a group *shared key* (SK) for efficient intra-group communications. However, having the same SK implies that all the group membership is at the same trust level. In a distributed, multicast group, it is often not possible nor desirable to have the *same* trust level throughout the group. One may be tempted to suggest that a single trust level can be defined by choosing the lowest possible trust level as the group trust level. Though such a straightforward approach is feasible, one can do better by compartmentalizing the group based on local trust levels (*Id.*). Such a compartmentalization inevitably least to *clustering* of a given group. Compartmentalization also helps in having a better control over the set of key management and distribution functions as noted in (*Id.*).

While the entities in each cluster may share a common trust level, it may be that the clusters are mutually suspicious and have only *partial* trust in each other. Thus, a mechanism is desired that permits mutually suspicious parties to come together to generate a shared key. In order to avoid involving (and potentially paying) a third party, it is also desirable that the scheme involve only the group members and not external parties.

Some schemes (such as Harney, H. and Muckenhirn, C., "GKMP Architecture," *RFC 2093* (1997); Harney, H. and Muckenhirn, C., "GKMP Architecture," *RFC 2094* (1997); Ballardie, A., "Scalable Multicast Key Distribution," *RFC 1949* (1996)) propose to replace the traditional (external) Key Distribution Center (KDC) with a *Group Controller* (GC) which can generate and distribute the keys. However, in these approaches, a single member is allowed to generate the keys. This means that group members must place complete trust in this group member. In (Poovendran, R., *et al.*, "A Scalable Extension of Group Key Management Protocol," *Proc. 2nd Ann. ATIRP Conf.*, Maryland, pp. 187-191 (1998)), a *panel* of members are allowed to generate the keys. However, this reference does not present any explicit distributed key generation scheme.

(Note: The following references are incorporated herein by reference: Bellare and Micali, "Non-Interactive Oblivious Transfer and Applications," in *Advances in Cryptology -- Crypto '89*, Springer-Verlag (1989), pp. 547-557; Poovendran *et al.*, "A Distributed Shared Key Generation Procedure Using Fractional Keys," *Proceedings of the MILCOM '98*, Boston, MA (Oct. 1998); Simmons, G.J., "An Introduction to Shared Secret and/or Shared Control Schemes and Their Applications," in *Contemporary Cryptology: The Science of Information Integrity*, Simmons, G.J., ed., IEEE Press (1992), pp. 441-497.)

Summary of the Invention

The invention described herein represents a new class of distributed key generation and recovery methods suitable for group communication systems where the group membership must be tightly controlled. The key generation approach allows entities which may have only partial trust in each other to jointly generate a shared key without the aid of an external third party. The group collectively generates and maintains a dynamic group binding parameter, and the shared key is generated using a pseudorandom function using this parameter as a seed. The methods employ distributed algorithms based on fractional keys (FK). The methods allow the members to automatically update the keys in a periodic manner without any assistance from an external third party, and to do so using verifiable secret sharing techniques. The key retrieval method does not require the keys to be stored in an external retrieval center. Note that many Internet-based applications may have these requirements.

Features and Advantages

The invention described herein has the feature of developing a shared key based on components associated with respective members of a cluster. The invention has the additional feature of a dynamic group binding parameter that

serves a seed for development of the shared key. The invention has the advantage of allowing cooperative key generation without requiring action by an independent party. The invention has the further advantage of allowing key retrieval without requiring the archiving of keys at an external retrieval center.

Brief Description of the Figures

The foregoing and other features and advantages of the invention will be apparent from the following, more particular description of a preferred embodiment of the invention, as illustrated in the accompanying drawings.

FIG. 1 is a flowchart illustrating the overall operation of an embodiment of the invention.

FIG. 2 is an example system implementing the invention.

FIG. 3 is a flowchart illustrating the initialization process as performed by a security manager, according to an embodiment of the invention.

FIG. 4 is a flowchart illustrating the initialization process as performed by cluster members in a distributed fashion, according to an embodiment of the invention.

FIG. 5 is a flowchart illustrating subsequent key generation, according to an embodiment of the invention.

FIG. 6 is a flowchart illustrating subsequent key generation using ElGamal public key pairs, according to an embodiment of the invention.

FIG. 7 is a flowchart illustrating key recovery, according to an embodiment of the invention.

FIG. 8 is a flowchart illustrating verification of security manager-based initialization, according to an embodiment of the invention.

FIG. 9 is a flowchart illustrating verification of distributed initialization, according to an embodiment of the invention.

FIG. 10 illustrates an example computing environment of the invention.

Detailed Description of the Preferred Embodiments

A preferred embodiment of the present invention is now described with reference to the figures where like reference numbers indicate identical or functionally similar elements. Also in the figures, the left most digit of each reference number corresponds to the figure in which the reference number is first used. While specific configurations and arrangements are discussed, it should be understood that this is done for illustrative purposes only. A person skilled in the relevant art will recognize that other configurations and arrangements can be used without departing from the spirit and scope of the invention. It will be apparent to a person skilled in the relevant art that this invention can also be employed in a variety of other devices and applications.

I. Properties of the Key Generation and Management Method

The following notation is used to describe the different entities involved in the method:

$\alpha_{i,j}$: The one-time pad of the *i*th member at the *j*th key update iteration.

θj: The group binding parameter at the *j*th key update iteration.

$\{K_i, K_i^{-1}\}$: Public key pair of the member *i*. This pair is assumed to be updated appropriately to preserve the integrity and confidentiality of any communication transaction by and with member *i*.

$FK_{i,j}$: The FK of the *i*th member at the *j*th key update iteration.

$HFK_{i,j}$: The *hidden* FK (HFK) of the *i*th member at the *j*th key update iteration.

SK_j: The group SK at the *j*th key update instance.

$A \rightarrow B:X$: Principal A sends principal B a message X.

In an embodiment of the invention, the message format is $\left\{\left\{T_i, M, j, Msg\right\}_{K_s^{-1}}\right\}_{K_R}$, where the variables are defined as follows:

- T_i: a real-valued, wallclock time stamp generated by member i
- M: denotes the *mode* of operation, with "I" for Initialization mode, "G" for Generation mode, and "R" for key Recovery mode.
- j: integer-valued, denotes the current iteration number.
- Msg: the message to be sent.
- K_s^{-1}: denotes the private key of the sender S.
- K_R: public key of the receiver R.

The following properties are desirable for a multiparty key generation scheme:

- An FK contributed by a participating member should have the same level of security as the group SK.
- A single participating member, without valid permissions, should not be able to obtain the FK of another member.
- If a FK-generating member has physically failed, been compromised or removed, the remaining FK-generating members should be able to jointly recover the FK of the failed member.

The first property simply states that the distributed key generation scheme has to be such that each FK space has at least the same size as the final SK space. Hence, each member may generate FK of different size but, when combined, they lead to a fixed length SK. The second property has to do with the need for protection of individual FKs that is desired in light of the absence of a centralized key generation scheme. In the current scheme, every member performs an operation to hide its FK such that, when all the hidden FKs (HFKs) and the group

parameter are combined, the net result is a new SK. Even if an HFK is known, the problem of obtaining the actual FK or the SK needs further computation. The requirements of the FK concealment mechanism are described in greater detail below.

If a contributing member physically fails, becomes compromised, or has to leave the multicast group, or cluster, then it becomes necessary to replace the existing member with a new member. Hence, the newly-elected member should be able to securely recover the FK generated by the replaced member. However, to ensure the integrity of the scheme, this recovery should be possible only if all the remaining contributing members cooperate. This feature deviates significantly from the existing key generating schemes (Harney, H. and Muckenhirn, C., "GKMP Architecture," *RFC 2093* (1997); Harney, H. and Muckenhirn, C., "GKMP Architecture," *RFC 2094* (1997); Ballardie, A., "Scalable Multicast Key Distribution," *RFC 1949* (1996)). The requirement that an individual member acting alone not be able to obtain the FKs of other contributing members is similar to protecting individual private keys in public key cryptography systems.

The following is a list of assumptions regarding the method:

- There exist two commutative operators ⊚ and ◇ which form an abelian group when operating on the set of keys.
- It is computationally difficult to perform cryptographic analysis on a cryptographically-secure random key by search methods if the key length is sufficiently large.
- The keys are all L bits in length, and all members know its length.
- The number of participants in generating the KS is fixed as n (where n may be a function of ⊚ and ◇).
- There is a mechanism for certifying the members participating in the key generation procedure, for securely exchanging the quantities required in the algorithm and for authenticating the source of these quantities.
- Every member has the capability to generate a cryptographically-secure random number of length L bits or longer.

With the assumptions above, the key management scheme can be described in terms of three major processes:

1. Initialization, which includes secure initial one-time pad and binding parameter generation and distribution;
2. Key Generation, an iterative process including fractional, hidden and shared-key generation; and
3. Key Retrieval, a process that is required only in the case of a member node failure or compromise.

These processes are collectively illustrated in process 100 of FIG. 1. Process 100 begins with a step 105. In a step 110, the key management process is initialized. Here, initial one-time pads are generated for each member. In addition, a binding parameter is generated and distributed to each member, permitting each member to generate the same key, a shared key SK. In a step 115, the members can operate securely using the SK. If, in a step 120, a failure occurs at a member's node, such as a compromise of the member or an equipment failure, then key retrieval is performed in a step 125. Here, recovery of the parameters associated with the failed node is performed. In a step 130, a new binding parameter is generated and new one-time pads are created. Operations then resume at step 115.

If, in step 120, no failure occurs, process 100 continues with a step 135. Here, a determination is made as to whether an update of the SK is needed. This may be required, for example, if a member leaves the cluster. Alternatively, an operation may simply require periodic updating of the SK. If an update is needed, key generation step 130 is performed. Operations then resume at step 115.

The processes of initialization, key generation, and key retrieval are described in greater detail below.

II. Initialization

A Group Initiator (GI) first selects a set of n FK-generating members of a cluster, and the GI may be one of these members. The GI can then contact a Security Manager (SM)–a third party who is *not* a FK-generating member–who generates the initial pads and the binding parameter and distributes them to the members. This is illustrated by system 200 of FIG. 2. Member 1, group initiator 210, is shown contacting security manager 250, who then distributes the necessary data to member 1 through 4, labelled 210 through 240, respectively. The data flow for this embodiment is illustrated by dotted lines. In an alternative embodiment, GI 210 initiates a distributed procedure among the group members (illustrated by solid lines) to create these quantities without the aid of an external party.

A. SM-Based Initialization

The process of initialization by an SM is illustrated in FIG. 3, process 300, according to an embodiment of the invention. Process 300 begins with a step 305. In a step 310, the GI generates an initial random one-time pad, $\alpha_{i,1}$, for each member i. In a step 315, an initial binding parameter θ_1 is computed such that $\alpha_{1,1} \circledcirc \alpha_{2,1} \circledcirc \ldots \circledcirc \alpha_{n,1} = \theta_1$. In steps 320 through 340, $\alpha_{i,1}$ and θ_1 are sent to each member i. In step 320, index i is initialized. In steps 325 and 330, the initial pads and binding parameter are distributed to member i, as

$$SM \rightarrow i: \left\{ \left\{ T_{SM}, I, 1, \alpha_{i,1}, \theta_1 \right\}_{K_{SM}^{-1}} \right\}_{K_i} .$$

In step 335, index i is incremented. In step 340, a determination is made as to whether $\alpha_{i,1}$ and θ_1 have been sent to all members i. If not, then $\alpha_{i,1}$ and θ_1 are sent to the next member i. The process concludes with a step 345. At the conclusion of process 300, each member has θ_1, on which a common SK can be based.

B. Distributed Initialization

In an alternative embodiment, initialization can be performed through a cooperative process involving all members, illustrated as process 400 of FIG. 4. The GI (assumed to be a member and denoted in process 400 by the index 1) can perform the following steps (see also Koblitz, N., *Cryptologia* 317-326 (1997), incorporated herein by reference) to generate the initial parameters of the group. Process 400 begins with a step 405 In a step 410, member 1 generates two uniformly-distributed random quantities γ and $v_{1,1}$ of bit length L. In a step 415, member 1 operates on these two quantities as $\gamma \circledcirc v_{1,1} = \delta_1$. In a step 420, member 1 sends the result to member 2 (the "next" member in the group) as $1 \rightarrow 2$: $\left\{\left\{T_1, I, 1, \delta_1\right\}_{K_1^{-1}}\right\}_{K_2}$.

Starting with member 2, each member i calculates its own δ_i based on the previous member's δ_{i-1}, and sends δ_i to the next member. This is illustrated in steps 425 through 450. In step 425, the index i is initialized to 2 In step 430, member i generates a uniform random variable $v_{i,1}$ of bit length L. In step 435, member i then operates on the quantity it received from member $i-1$ as $\delta_{i-1} \circledcirc v_{i,1} = \delta_i$. In step 440, member i then sends the result to member $i+1$ as $i \rightarrow i+1$: $\left\{\left\{T_i, I, 1, \delta_i\right\}_{K_i^{-1}}\right\}_{K_{i+1}}$.

In step 445, i is incremented. If, as determined in step 450, each of the n members has not generated a respective value δ_i, the process returns to step 430, where the next member i generates its uniform random variable $v_{i,1}$.

Eventually, the group member $i = n$ receives δ_{n-1} and, in a step 455, generates a uniformly-distributed random quantity $v_{n,1}$ of bit length L. In a step 460, member n performs $\delta_{n-1} \circledcirc v_{n,1} = \delta_n$. In a step 470, member n securely sends δ_n to the initiating member $i = 1$ as $n \rightarrow 1$: $\left\{\left\{T_n, I, 1, \delta_n\right\}_{K_n^{-1}}\right\}_{K_1}$. In a step 475,

the GI (member 1) then recovers δ_n and performs $\gamma \circledcirc \delta_n = \theta_1$. In steps 480 through 494, member 1 sends θ_1 to each member *i*. In step 480, the index *i* is initialized to 2. In step 485, member 1 sends θ_1 to member i as

$$1 \to i: \left\{\left\{T_1, I, 1, \theta_1\right\}_{K_1^{-1}}\right\}_{K_i}.$$

In step 490, each member *i* privately computes $\alpha_{i,1} = \theta_1 \circledcirc \nu_{i,1}$ In step 492, the index *i* is incremented. If, in step 494, i > n, so that each member *i* has received θ_1 and privately computed a respective $\alpha_{i,1}$, then the process 400 concludes with a step 496. Otherwise, the process returns to step 485, where member 1 sends θ_1 to another member. At the conclusion of process 400, each member has θ_1, on which a common SK can be based.

Note that these two approaches of initialization (security manager-controlled initialization and distributed initialization) are not equivalent unless additional security assumptions are made. For example, in the case of distributed initialization within the group, the following can be done.

Assume that members $i-1$ and $i+1$ conspire to obtain the secret member *i*, where the numerical ordering corresponds to the order of message passing in the distributed algorithm.

1. Member $i-1$ sends δ_{i-1} to member *i* as per the algorithm, and *also* to member $i+1$ without *i*'s knowledge.
2. Member *i*, who is unaware of the conspiracy between $i-1$ and $i+1$, computes $\delta_i = \delta_{i-1} \circledcirc \nu_{i,1}$ and sends it to member $i+1$ securely.
3. Member $i+1$ can now compute $\nu_{i,1} = \delta_{i-1} \circledcirc \delta_i$ and obtain the secret $\nu_{i,1}$ of member *i*.

However, the secret vi,1 generated by member *i* become part of the pads (i.e. the α's) of members $i-1$ and $i+1$. Hence, application of this initialization assumes that the parties are benign.

III. Key Generation

The key generation algorithm is an iterative process depicted in FIG 5 as process 500. Each successive key generation, iteration *j*, requires as input a set of one-time pads $\alpha_{i,j}$, $i = 1, \ldots, n$, and the binding parameter θ_j, which are obtained from the initialization process (e.g., process 300 or process 400) for iteration $j = 1$, and from the preceding iterations for $j > 1$.

The iterative key generation process, according to an embodiment of the invention, consists of the following. Process 500 begins with a step 505. In steps 510 through 535, each member *i* generates a cryptographically-secure random number, fractional key $FK_{i,j}$, and sends it to every other member *m*. In step 510, index *i* is initialized to 1. In step 515, member *i* generates random number $Fk_{i,j}$. In step 520, member *i* generates a hidden fractional key $HFK_{i,j} = \alpha_{i,j} \circledcirc FK_{i,j}$. In step 525, member *i* sends $HFK_{i,j}$ to every other member m as

$$i \rightarrow m: \left\{ \left\{ T_i, G, j, HFK_{i,j} \right\}_{K_i^{-1}} \right\}_{K_m}.$$

In step 530, index *i* is incremented. If, as determined in step 535, each member *i* has created a respective $HFK_{i,j}$ and sent it to all other members, the process continues at a step 540. Otherwise, process 500 returns to step 515, where the next member *i* generates its respective $FK_{i,j}$.

Once the exchange of $HFK_{i,j}$'s is complete, each member computes the new group parameter θ_{j+1} and a new shared key SK_j. This occurs in steps 540 through 560. In step 540, index *i* is initialized to 1 In step 545, member *i* calculates the new binding parameter, $\theta_{j+1} = \lambda\theta_j \circledcirc HFK_{1,j} \circledcirc HFK_{2,j} \circledcirc \ldots \circledcirc HFK_{n,j} = FK_{1,j} \circledcirc FK_{2,j} \circledcirc \ldots FK_{n,j}$. In step 550, member *i* calculates a new one-time pad $\alpha_{i,j+1} = \theta_{j+1} \circledcirc FK_{i,j}$, and a new shared key $SK_j = f(\theta_{j+1})$ where $f(\cdot)$ is a strong one-way pseudo-random function. In step 555, index *i* is incremented. If, in step 560, $i > n$, so that each member *i* has created a new θ_{j+1} and a new SK_j, then the

process concludes with a step 565. Otherwise, process 500 returns to step 545, where the next member i calculates the new binding parameter, θ_{j+1}.

If the resulting group parameter θ_{j+1} is cryptographically insecure for a particular application, all members can repeat process 500 creating a new high quality group parameter θ_{j+1}.

At the end of process 500, we have the SK for the current iteration Note that the quantity $\alpha_{i,j+1}$ is computed such that, for an outsider, obtaining $\alpha_{i,j+1}$ is very hard, even if the actual shared key SK_j is compromised at any key update time interval $(j, j+1)$. Knowing the shared key SK_j does not reveal the group parameter θ_j and, hence, the tight binding of the members will not be broken by the loss of the shared key.

Note the following additional features of the key scheme:

- Although all the members have each $HFK_{i,j}$, obtaining the $FK_{i,j}$ or $\alpha_{i,j+1}$ of another member involves search in the L-dimensional space, and obtaining their correct combination involves search in the $(n-1)L$ - dimensional space. Hence, even if a fellow member becomes an attacker, that rogue member faces nearly the same computational burden in obtaining the set of n FKs as an outside cryptographic analyst; i.e. trust is *not* unconditional.
- For such an outside attacker, breaking the system requires either search in an L-dimensional space to get θ, or nL - dimensional searches to break individual secrets of all the members. Access to all n HFKs is alone is insufficient to permit an attacker to determine the SK; for that, the attacker must also possess the current binding parameter θ which is time-varying and never transmitted. If an SK is known to be compromised (perhaps due to traffic analysis), information regarding θ is not obtained , since $f(\cdot)$ is a pseudo-random function.

In an embodiment of the invention, an $FK_{i,j}$ is used whereby $(FK_{i,j}^{-1}, FK_{i,j})$ is an individual ElGamal public key pair for the member i at update j. The iterative key generation process for this embodiment is illustrated as process 600 of FIG. 6. Process 600 begins with a step 605. In steps 610 through

640, each member i develops values $FK_{i,j}$ and $HFK_{i,j}$ and exchanges them with other members. In step 610, index i is initialized to 1. In step 615, member i randomly picks a number $FK_{i,j}^{-1}$ with $0 \le FK_{i,j}^{-1} \le p-2$. In step 620, member i generates $FK_{i,j} = \alpha^{FK_{i,j}^{-1}}$. Here, $(FK_{i,j}^{-1}, FK_{i,j})$ is an individual ElGamal public key pair for the member i at update j In step 625, member i generates a quantity $HFK_{i,j} = (\alpha_{i,j} + FK_{i,j}) \bmod p$ In step 630, member i sends $FK_{i,j}$ and $HFK_{i,j}$ to each other member m, in the form

$$i \rightarrow m: \left\{ \left\{ T_i, G, j, HFK_{i,j}, FK_{i,j} \right\}_{FK_{i,j-1}^{-1}} \right\}_{FK_{m,j-1}}$$

. In step 635, index i is incremented. If, as determined in step 640, $i > n$, so that each member i has created a respective $HFK_{i,j}$ and sent it, along with $FK_{i,j}$, to all other members, the process continues at a step 645. Otherwise, process 600 returns to step 615, where the next member i selects its respective $FK_{i,j}^{-1}$.

In steps 645 through 665, each member generates a new binding parameter θ_{j+1} and one-time pad $\alpha_{i,j+1}$. In step 640, index i is initialized to 1. In step 650, each member i computes $\theta_{j+1} = \left((p-n-3)\theta_j + \sum_{i=1}^{i=n} HFK_{i,j} \right) \bmod (p-1)$, defining $GK_{j+1}^{-1} = \theta_{j+1}$. Each member i also computes $GK_{j+1} = \alpha^{GK_{j+1}^{-1}} = \prod_{i=1}^{i=n} FK_{i,j} = \prod_{i=1}^{i=n} \alpha^{FK_{i,j}^{-1}}$ in step 650. In step 655, member i calculates $a_{i,j+1} = (GK_j^{-1} + FK_{i,j}^{-1}) \bmod p$. In step 660, index i is incremented. In step 665, a determination is made as to whether $i > n$, i.e., whether each member i has calculated the new θ_{j+1} and a new $\alpha_{i,j+1}$. If so, process 600 concludes with a step 670. Otherwise, process 600 returns to step 650 so that the next member i can create a new θ_{j+1}.

Note that if the resulting group key pair $(GK_{J+1}, GK_{J+1}^{-1})$ is cryptographically insecure for a particular application, all members can repeat process 600, creating a new high quality key pair.

IV. Retrieval of the Fractional Key and One-time Pad of a Failed Node

The following steps, illustrated as process 700 of FIG. 7, are involved in recovery of the $FK_{i,j}$ and $\alpha_{i,j}$ of the node failed i, where j represents the iteration number in which the node was compromised or failed. The process begins with a step 705. In a step 710, any one FK-generating member–called the Recovery Initiator (RI)– initiates recovery and gives the HFK of the failed node i to the newly-elected node i as $RI \rightarrow i$: $\left\{\left\{T_{RI}, R, j, HFK_{i,j}\right\}_{K_{RI}^{-1}}\right\}_{K_i}$. In a step 615, the RI gives the newly-elected node i the current SK_j as $RI \rightarrow i$. $\left\{\left\{T_{RI}, R, j, SK_j\right\}_{K_{RI}^{-1}}\right\}_{K_i}$. In a step 720, distributed initialization is performed, with the following replacements: (a) θ by ξ and (b) $\alpha_{l,j}$ by $\beta_{l,j}$. Except for the changes in the notation and the number of members participating, the process for pad generation is same as for distributed initialization. Hence, at the end of this distributed pad generation, each member l has $\beta_{l,j}$ as its pad for key recovery process, and all these pads are bound with the parameter ξ. In steps 725 through 745, each member l calculates a modified hidden fractional key $\hat{HFK}_{l,j}$ and distributes it to newly elected member i. In step 725, index l is initialized to 1. In step 730, member l computes modified hidden fractional key $\hat{HFK}_{l,j} = \beta_{l,j} \diamond FK_{l,j}$ and sends it to the newly-elected member i as $l \rightarrow i$: $\left\{\left\{T_l, R, j, \hat{HFK}_{l,j}\right\}_{K_l^{-1}}\right\}_{K_i}$ in step 735. In step 740, index l is incremented. In step 745, a determination is made as to whether l > n, i.e.,

whether each member l has calculated a modified hidden fractional key $\hat{HFK}_{l,j}$ and distributed it to newly elected member i. If not, process 700 returns to step 730. Otherwise, process 700 continues with a step 750.

In step 750, member i combines all of the modified HFKs and recovers the fractional key $FK_{i,j}$ using the operation $FK_{i,j} = \lambda\xi \circledcirc \hat{HFK}_{1,j} \circledcirc \ldots \circledcirc \hat{HFK}_{i,j} \circledcirc \theta_{j+1}$. In step 755, member i extracts the one-time pad $\alpha_{i,j}$ using the operation $\alpha_{i,j} = HFK_{i,j} \circledcirc FK_{i,j}$. The process 700 concludes with a step 760.

Note that the recovered values of $FK_{i,j}$ and $\alpha_{i,j}$ are unique. Once the new node recovers the fractional key of the compromised node, it can inform the other contributing members to update the iteration number j to $j + 1$, and then all members can execute the key generation algorithm. Note that even though the newly-elected member recovers the compromised fractional key and pad, the next key generation operation of the new node does not use the compromised key or pad. Hence, even if the attacker possesses the fractional key or pad at iteration j, it does not allow the attacker to obtain the future fractional keys or pads without any computation.

V. A Specific Choice of the Functions ⊚ **and** ◇

A class of multiparty key generation algorithms is described above where a given instance of the class is determined by choice of function ⊚. Note that one possible choice for ⊚ is the modulo addition operation with respect to a large odd prime p, denoted here with $\oplus$. In this case, we can deduce the following computation from the key generation algorithm:

$$HFK_{1,J} \oplus HFK_{2,J} \oplus \cdots \oplus HFK_{n,J} =$$
$$FK_{1,J} \oplus FK_{2,J} \oplus \cdots \oplus FK_{n,J} \oplus (n-1)\theta_J$$

To remove the effect of θ_j on θ_{j+1}, we should ensure that $\lambda = (p + 1 - n)$ so that

$$\theta_{J+1} = (p+1-n)\theta_J \oplus HFK_{1,J} \oplus HFK_{2,J} \oplus \cdots$$
$$\cdots \oplus HFK_{n,J}$$
$$= FK_{1,J} \oplus FK_{2,J} \oplus \cdots \oplus FK_{n,J}$$

Regarding the choice of the number of members, clearly the choice of $n = 2$ is not appropriate for such a scheme. Although choosing $n = 3$ does not instantly expose a secret pad α; when a participating member becomes an attacker (i.e. a *rouge*), the following attack–called *fractional attack* (FA)–is feasible.

Lemma: When $\circledcirc$ is an $\oplus$ function, independent of how nontrivial the bit-length of the key is, choosing $n = 3$ permits a FA.

Proof: Assume that the time instant at which one member i ($i = 1$ or 2 or 3) become a *rogue* is j. At this time the member have values of $\alpha_{1,J} = HFK_{2,J} \oplus HFK_{3,J}$, $\alpha_{2,J} = HFK_{3,J} \oplus HFK_{1,J}$, $\alpha_{3,J} = HFK_{1,J} \oplus HFK_{2,J}$. Every member also has access to the current θ_{j+1} and their own $FK_{l,J}$ ($l = 1, 2, 3$). At this stage, obtaining the α component of any other member is as computationally intensive as an outside attacker trying to obtain θ_{j+1}. However, if a member, say $\imath = 1$, is compromised and releases its secret $\alpha_{1,J}$, then each of the other members can use this and compute $FK_{1,J} = \alpha_{1,J} \oplus \theta_J$. Since $\theta_{J+1} = FK_{1,J} \oplus FK2_{2,J} \oplus FK_{3,J}$, each member can now compute the other non-rogue member's FK as well.

This leads to the following corollary: When $\circledcirc$ is an $\oplus$ function, independent of how non-trivial the bit-length of the key, the minimum number of members to prevent a FA by a single *rogue* member for the multiparty key scheme is 4.

VI. Verifiable Secret Sharing

Since there are multiple entities involved in key generation, it becomes important to have a mechanism to verify if the parameters exchanged actually contribute to the generated shared key. The verification steps can be followed at (1) SM-based group initialization, (b) distributed group initialization, and (c) θ-generation iteration.

A. SM-based Initialization

In the case of the SM-based scheme, each member *i* needs to make sure that the SM uses non-trivial values of its $\alpha_{i,1}$ and θ_1. Since each member needs to protect its individual pad value, one method for openly checking correctness of the pads is to generate a public value that will enable all the key generating members to check their correctness without revealing the actual value of the individual pads. Such a verification technique falls under the category of Verifiable Secret Sharing (VSS) (Feldman, P., "A Practical Scheme for Non-Interactive Verifiable Secret Sharing," *Proc. of IEEE Fund. Comp. Sci.*, pp. 427-437 (1987); Pedersen, T. P., *Advances in Cryptology - CRYPTO, LNCS 576*:129-140 (1991)).

If one wants to check if the individual initial pads $\alpha_{i,1}$ given by the security manager are "good", process 800 of FIG. 8 can be used. The process begins with a step 805. In a step 810, one member (possibly the SM) picks a very large prime number q. The number picked should be larger than the possible range of the θ value. In a step 820, prime number q is sent to all the members. In a step 825, the same member also sends a generator g of the multiplicative group q. In a step 830, each member picks a random polynomial f_i having a value 0 at the origin. In a step 835, each member adds the polynomial to its pad value, generates $\hat{\alpha}_{i,1} = g^{\alpha_{i,1}+fi}$ and broadcasts the values to all the members. In a step 840, each member i computes $g^{\theta_1} = \prod_{j=1}^{j=n} \hat{\alpha}_{i,1} = g^{\theta_1}$. In a step 845, each member

checks if the value is equal to g^{θ_1} at the origin. If not, then the verification fails in a step 850. If the check of step 845 passes, then in a step 855, each member checks to see that

$$g^{\alpha_{i,1}} = \frac{g^{\theta_1}}{\prod_{j=1}^{j=n} g^{\alpha_{j,1}}}$$

If not, verification fails in step 850. Failed verification means that some or all of the members' one-time pads do not correspond to θ_1. Process 800 concludes with a step 860.

B. Distributed Initialization

In the case of distributed initialization, process 900 of FIG. 9 can be used to check if the GI, member 1, has produced a θ_1 using contributions from all the group members. The process begins with a step 905. In a step 910, one member (possibly the GI) picks a very large prime number q. The number picked should be larger than the possible range of the θ_1 value. In a step 915, prime number q is sent to all the members. In a step 920, the same member also sends a generator g of the multiplicative group under q to all members. In a step 925, GI computes g^{γ} and $g^{v_{1,2}}$, and makes them available to all the group members. In a step 930, each member i publishes $g^{v_{i,1}}$ making it available only to the group members. In a step 935, each member i checks if $g^{\theta_1} = \prod_{j=1}^{j=n} g^{v_{j,1}}$. If the equality is not true, then failed verification is indicated in a step 940. Failure (inequality) means that the binding parameter θ_1 and the individual one-time pads do not agree. Since at each step of adding their secrets members published the broadcast values, it is possible to check which member cheated if there is no collaboration. If there

is a collaboration, then the last among the collaborating member can be identified by the non-collaborating member.

Note that similar testing can be done for the key generation process.

VII. Environment

The present invention may be implemented using hardware, software or a combination thereof. The operations described above may be implemented in a computer system or other processing system at the node of a member. An example of such a computer system 1000 is shown in FIG. 10. The computer system 1000 includes one or more processors, such as processor 1004. The processor 1004 is connected to a communication infrastructure 1006, such as a bus or network). Various software implementations are described in terms of this exemplary computer system. After reading this description, it will become apparent to a person skilled in the relevant art how to implement the invention using other computer systems and/or computer architectures.

Computer system 1000 also includes a main memory 1008, preferably random access memory (RAM), and may also include a secondary memory 1010. The secondary memory 1010 may include, for example, a hard disk drive 1012 and/or a removable storage drive 1014, representing a floppy disk drive, a magnetic tape drive, an optical disk drive, etc. The removable storage drive 1014 reads from and/or writes to a removable storage unit 1018 in a well known manner. Removable storage unit 1018, represents a floppy disk, magnetic tape, optical disk, or other storage medium which is read by and written to by removable storage drive 1014. As will be appreciated, the removable storage unit 1018 includes a computer usable storage medium having stored therein computer software and/or data.

In alternative implementations, secondary memory 1010 may include other means for allowing computer programs or other instructions to be loaded into computer system 1000. Such means may include, for example, a removable

storage unit 1022 and an interface 1020. Examples of such means may include a program cartridge and cartridge interface (such as that found in video game devices), a removable memory chip (such as an EPROM, or PROM) and associated socket, and other removable storage units 1022 and interfaces 1020 which allow software and data to be transferred from the removable storage unit 1022 to computer system 1000.

Computer system 1000 may also include a communications interface 1024. Communications interface 1024 allows software and data to be transferred between computer system 1000 and external devices. Examples of communications interface 1024 may include a modem, a network interface (such as an Ethernet card), a communications port, a PCMCIA slot and card, etc. Software and data transferred via communications interface 1024 are in the form of signals 1028 which may be electronic, electromagnetic, optical or other signals capable of being received by communications interface 1024. These signals 1028 are provided to communications interface 1024 via a communications path (i.e., channel) 1026. This channel 1026 carries signals 1028 and may be implemented using wire or cable, fiber optics, a phone line, a cellular phone link, an RF link and other communications channels. In an embodiment of the invention in which computer system 1000 represents the computer system of a member's node, signals 1028 comprise information flowing to and from the node, such as the encrypted form of δ_i in step 440, and the encrypted form of $HFK_{i,j}$ of step 525.

In this document, the terms "computer program medium" and "computer usable medium" are used to generally refer to media such as removable storage units 1018 and 1022, a hard disk installed in hard disk drive 1012, and signals 1028. These computer program products are means for providing software to computer system 1000.

Computer programs (also called computer control logic) are stored in main memory 1008 and/or secondary memory 1010. Computer programs may also be received via communications interface 1024. Such computer programs, when executed, enable the computer system 1000 to implement the present invention

as discussed herein. In particular, the computer programs, when executed, enable the processor 1004 to implement the present invention. Accordingly, such computer programs represent controllers of the computer system 1000. Where the invention is implemented using software, the software may be stored in a computer program product and loaded into computer system 1000 using removable storage drive 1014, hard drive 1012 or communications interface 1024. In an embodiment of the present invention, the steps of processes 300 through 900 are implemented in software that can therefore be made available to processor 1004 at a member node through any of these means.

VIII. Conclusion

While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example, and not limitation. It will be apparent to persons skilled in the relevant art that various changes in detail can be made therein without departing from the spirit and scope of the invention. Thus the present invention should not be limited by any of the above-described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.

What Is Claimed Is:

1. A method of generating and managing shared keys for a plurality of members of a cluster, comprising the steps of

(a) system initialization to produce a functionally generated initial shared key;

(b) functional generation of a next shared key; and

(c) key recovery in the event of either compromise or failure of a node.

2. The method of claim 1, wherein step (a) comprises the steps of:

(i) generating a random initial one-time pad $\alpha_{1,2}$ for each member;

(ii) calculating an initial binding parameter θ_1 based on each $\alpha_{1,2}$, where $\theta_1 = \alpha_{1,1} \circledcirc \alpha_{2,1} \circledcirc - - - \circledcirc \alpha_{n,1}$ wherein $\circledcirc$ is a commutative operator; and

(iii) sending θ_1 and $\alpha_{i,1}$ to each member *i*.

3. The method of claim 2, wherein step (iii) comprises the step of encrypting θ_1 and $\alpha_{i,1}$ in the form

$$\left\{\left\{T_{SM}, I, 1, \alpha_{i,1}\right\}_{K_{SM}^{-1}}\right\}_{K_i}$$

for transmission to each member *i*, where

T_{SM} is a timestamp generated by a security manager (SM),

I is an indicator of an initialization mode,

1 denotes the first interaction of key generation,

K_{sm}^{-1} is an encryption operation using a private component of a private/public key pair of the security manager, and

K_i indicates encryption using a public component of a private/public bey pair of member i.

4. The method of claim 1, wherein step (a) comprises the steps of:

(i) generation, by a member 1, of random quantities γ and $\nu_{1,1}$;

(ii) calculation by the member 1, of $\gamma \circledcirc \nu_{1,1}=\delta_1$, wherein $\circledcirc$ is a commutative operator;

(iii) sending, by the number 1, of δ_2 to a member 2;

(iv) receipt, by a member i, of δ_{i-1} from a preceding member i-1;

(v) generation, by member i, of random quantity $\nu_{i,1}$;

(vi) calculation, by member i, of $\delta_{i-1} \circledcirc \nu_{i,1}=\delta_i$;

(vii) sending, by member i, of δ_i to a member i+1;

(viii) sending, by a last member n, of δ_n to member 1;

(ix) calculation, by member 1, of $\gamma \circledcirc \delta_n=\theta_1$;

(x) sending, by member 1, of θ_1 to each member;

(xi) calculation, by each member, of $\theta_1 \circledcirc \nu_{i,1} = \alpha_{i,1}$.

5. The method of claim 4, wherein step iii) comprises the step of encrypting δ_1 in the form

$$\left\{\left\{T_1, I, 1, \delta_1\right\}_{K_1^{-1}}\right\}_{K_2}$$ for transmission to member 2,

step (vi) comprises the step of encrypting δ_i in the form

$$\left\{\left\{T_i, I, 1, \delta_i\right\}_{K_i^{-1}}\right\}_{K_{i+1}}$$ for transmission to member i+1,

step (vii) comprises the step of encrypting δ_n in the form

$$\left\{\left\{T_n, I, 1, \delta_n\right\}_{K_n^{-1}}\right\}_{K_1}$$ for transmission to member 1, and

step (ix) comprises the step of encrypting θ_1 in the form $\left\{\left\{T_1, I, 1, \theta_1\right\}_{K_1^{-1}}\right\}_{K_i}$ for transmission to member *i*.

6. The method of claim 1, wherein step (b) comprises the steps of:

(i) generation, by each member *i*, of a cryptographically secure random number, $Fk_{i,j}$, where *j* denotes the key generation iteration;

(ii) calculation, by each member *i*, of $HFK_{i,j} = \alpha_{i,j} \circledcirc FK_{i,j}$, where $\circledcirc$ is a commutative operator;

(iii) sending, by each member *i*, of $HFK_{i,j}$ to each other member;

(iv) calculation, by each member *i*, of

$$\theta_{j+1} = \lambda\theta_j \circledcirc HFK_{1,j} \circledcirc HFK_{2,j} \circledcirc \cdots \cdots HFK_{n,j}$$

where λ is a scaling factor and *n* is the number of members in the cluster;

(v) calculation, by each member *i*, of

$$\alpha_{i,j+1} = \theta_{j+1} \circledcirc FK_{i,j}$$

(vi) calculation, by each member *i*, of a shared key

$$SK_{j+1} = f(\theta_{j+1})$$

where f is a strong one way function, to form a fractionally generated next shared key.

7. The method of claim 6, wherein the step (iii) comprises the step of encrypting $HFK_{i,j}$ in the form $\left\{\left\{T_i, G, j, HFK_{i,J}\right\}_{K_i^{-1}}\right\}_{K_m}$ for transmission to each other member m.

8. The method of claim 6, wherein

step (i) comprises the steps of:

(A) random selection, by each member i, of a number $FK_{i,j}^{-1}$,

where $0 \leq FK_{i,j}^{-1} \leq p\text{-}2$, wherein p is a large odd prime number, such that p-1 has large prime factors; and

(B) calculation, by each member i, of

$$FK_{i,j} = \alpha_{i,j}^{FK_{i,j}^{-1}};$$

step (ii) comprises the step of calculation, by each member i, of $HFK_{i,j} = (\alpha_{i,j} + FK_{i,j}) \bmod p$;

step (iii) comprises the step of encrypting, by each member i, of $HFK_{i,j}$ in the form

$$\left\{\left\{T_i, G, j, HFK_{i,j}, FK_{i,j}\right\}_{FK_{i,j-1}^{-1}}\right\}_{FK_{m,j-1}}$$

for transmission to each other member m;

step iv) comprises the step of calculating, by each member i, of

$$\theta_{j+1} = ((p\text{-}n\text{-}3)\ \theta_j + \sum_{i=1}^{i=n} HFK_{i,j}) \bmod (p\text{-}1)$$

$$= GK_{j+1}^{-1}; \text{ and}$$

step (v) comprises the step of calculation, by each member i, of

$$\alpha_{i,j+1} = (GK_{j+1}^{-1} + FK_{i,j}^{-1}) \bmod p.$$

9. The method of claim 1, wherein step c) comprises the steps of:

(i) sending, by a recovery initiator RI, of the hidden fractional key of a failed node $\bar{i}$, $HFK_{\bar{i},j}$, to a newly elected member i, where j represents the iteration in which node $\bar{i}$ failed;

(ii) sending, by RI, of SK_j to member i;

(iii) performing a distributed initialization process, so that each member l receives a binding parameter ξ and a random pad $\beta_{l,j}$;

(iv) calculation, by each member l, of $HFK_{l,j} = \beta_{l,j} \diamond FK_{l,j}$, where $\diamond$ is a commutative operator;

(v) sending, by each member l, of $HFK_{l,j}$ to member i;

(vi) calculation, by number i, of

$FK_{\bar{i},j} = \lambda\, \xi \diamond HFK_{1,j} \circledcirc \cdots \circledcirc HFK_{n-1,j} \circledcirc \theta_{j+1}$, where $\circledcirc$ is a commutative operator; and

(vii) calculation, by member i, of

$$\alpha_{\bar{i},j} = HFK_{\bar{i},j} \circledcirc FK_{\bar{i},j}$$

10. The method of claim 9, wherein

step (i) comprises the step of encrypting $HFK_{\bar{i},j}$ in the form

$\left\{\left\{T_{RI}, R, j, HFK_{\bar{i},j}\right\}_{K_{RI}^{-1}}\right\}_{K_I}$ for transmission to member i, where R indicates recovery mode;

step (ii) comprises the step of encrypting SK_j in the form

$\left\{\left\{T_{RI}, R, j, SK_j\right\}_{K_{RI}^{-1}}\right\}_{K_I}$ for transmission to member i; and

step (v) comprises the step of encrypting $HFK_{l,k}$ in the form

$\left\{\left\{T_I, R, j, HFK_{l,j}\right\}_{K_I^{-1}}\right\}_{K_I}$.

11. The method of claim 2, further comprising the step of

(d) verifying that each of initial pad $\alpha_{i,j}$ has contributed to the calculation of θ_1, performed after step (a).

12. The method of claim 11, wherein step (d) comprises the steps of:

(i) selection, by a predetermined member of the cluster, of a large prime q;

(ii) distribution of q to all members;

(iii) selection, by the predetermined member, of a generator g of the multiplicative group under q;

(iv) distribution of g to all members;

(v) selection by each member i, of a random polynomial f_i having a value of zero at the origin;

(vi) calculation, by each member i, of $\hat{\alpha}_{i,j} = g^{\alpha_{i+1}+f_i}$;

(vii) sending, by each member i, of $\hat{\alpha}_{i,1}$ to all other members;

(viii) calculation, by each member i, of

$$g^{\hat{\theta}_1} = \prod_{j=1}^{j=n} \hat{\alpha}_{i,j} = g^{\theta_1 + \sum_{j=1}^{j=n} f_i}$$

, evaluated at the origin;

(ix) determination, by each member i, of whether $g^{\theta_1} = g^{\hat{\theta}_1}$, evaluated at the origin; and

(x) determination, by each member i, of whether

$$g^{\alpha_{i,j}} = \frac{g^{\theta_1}}{\prod_{j=1}^{j=n} g^{\alpha_{j,1}}}.$$

13. The method of claim 4, further comprising the step of:

(e) verifying that each initial pad $\alpha_{i,1}$ has contributed to the calculation of θ_1, performed after step (a).

14. The method of claim 12, wherein step (e) comprises the steps of:

(i) selection, by a predetermined member of the cluster, of a large prime q,

(ii) distribution of q to all members;

(iii) selection, by the predetermined member, of a generator g of the multiplicative group under q;

(iv) distribution of g to all members;

(v) calculation, by member 1, of g^{Y} and $g^{v_{1,1}}$;

(vi) making g^{Y} and $g^{v_{1,1}}$ available to all members;

(vii) calculation, by each member i, of $g^{v_{i,1}}$;

(viii) publication, by each member i, of $g^{v_{i,1}}$ for other members of the cluster only;

(ix) determination, by each member i, of whether $g^{\theta_i} = \prod_{j=1}^{j=n} g^{v_{i,1}}$.

15. A system for generating and managing shared keys for a plurality of members of a cluster, comprising

initialization means for performing system initialization to produce a fractionally generated initial shared key;

fractional generation means for fractional generation of a next shared key; and

recovery means for performing key recovery in the event of either compromise or failure of a node.

16. A computer program product comprising a computer usable medium having computer readable program code that executes on a computer that participates in the generation and management of shared keys for a plurality of members of a cluster, said computer readable program code comprising:

(a) first computer readable program code logic for causing the computer to participate in system initialization, wherein the initialization produces a fractionally generated initial shared key;

(b) second computer readable program code logic for causing the computer to participate in the fractional generation of a next shared key; and

(c) third computer readable program code logic for causing the computer to participate in key recovery in the event of either compromise of failure of a node.

SUBSTITUTE SHEET (RULE 26)

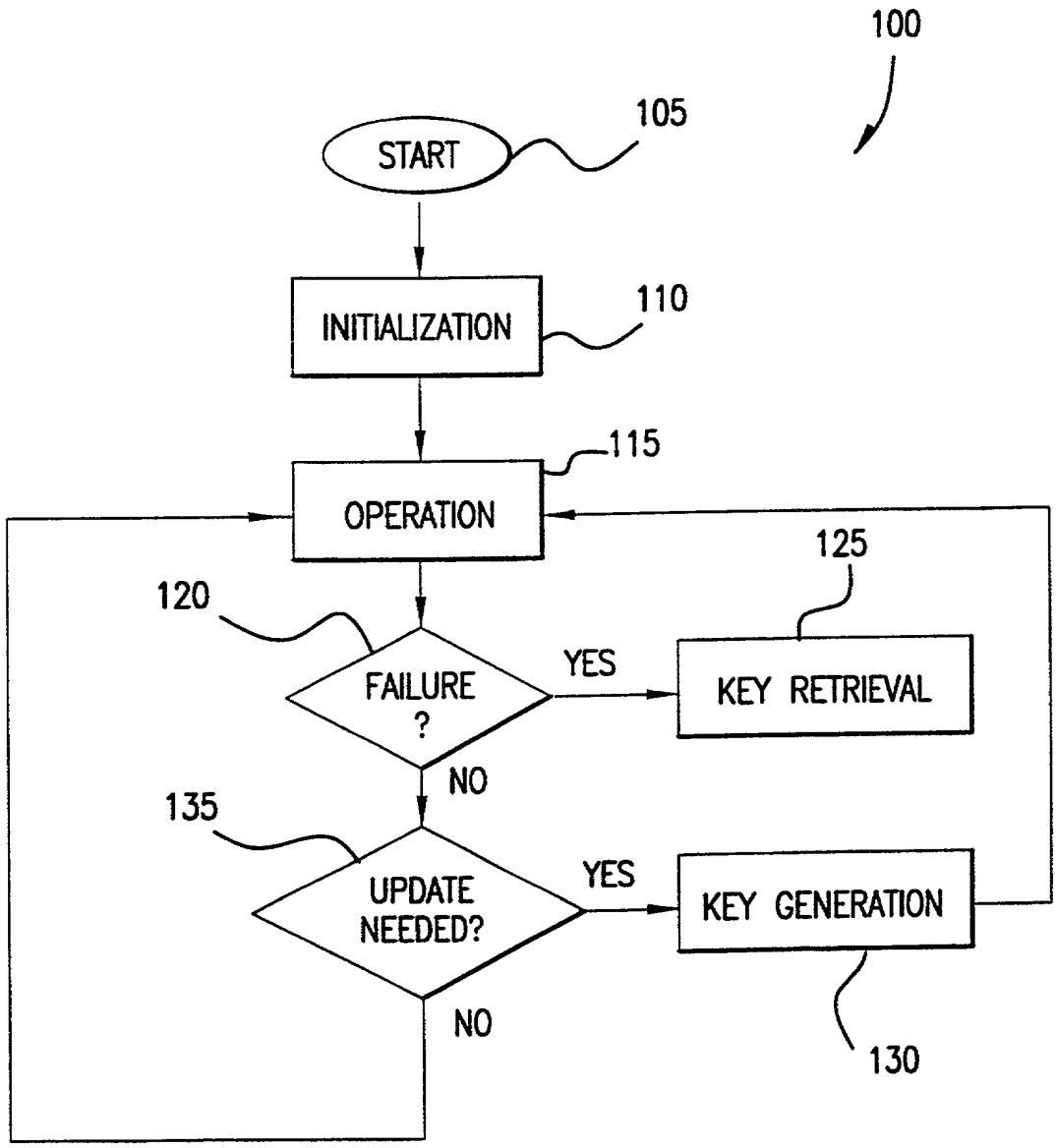
100
105
START
INITIALIZATION
110
115
OPERATION
125
120
FAILURE ?
YES
KEY RETRIEVAL
NO
135
UPDATE NEEDED?
YES
KEY GENERATION
NO
130

FIG.1

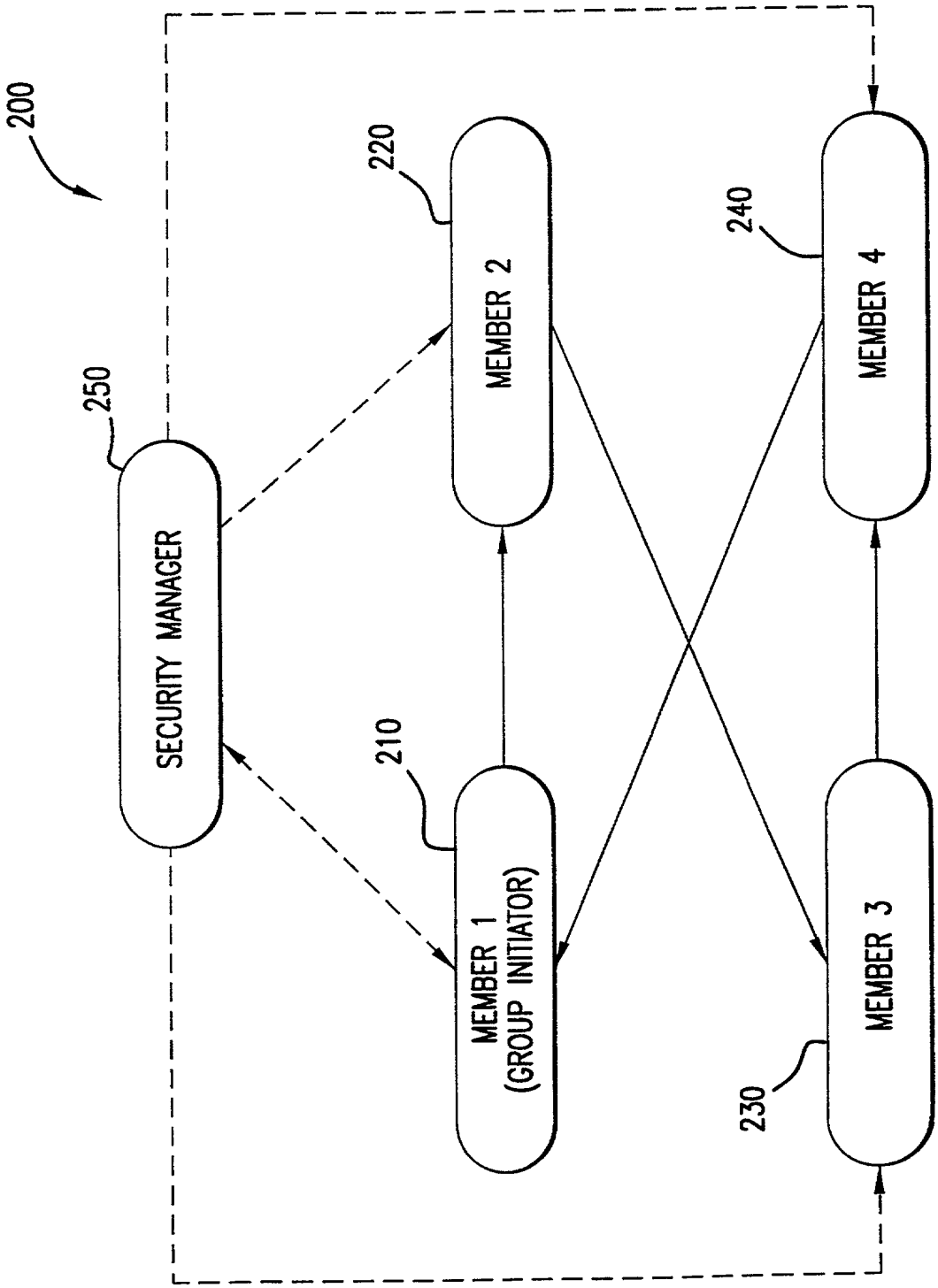
200
220
MEMBER 2
240
MEMBER 4
250
SECURITY MANAGER
210
MEMBER 1
(GROUP INITIATOR)
230
MEMBER 3

FIG.2

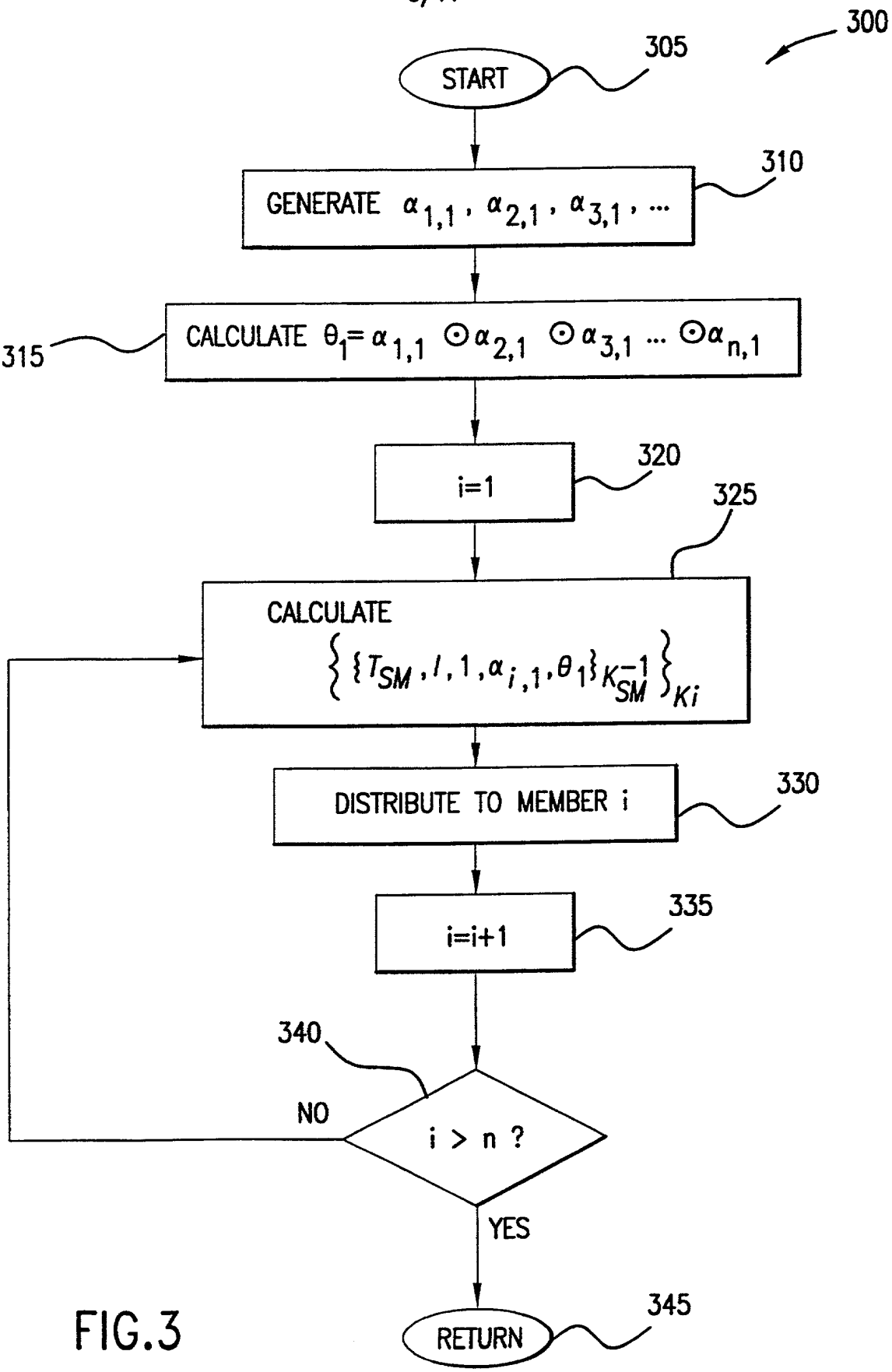
300
305
START
310
GENERATE $\alpha_{1,1}, \alpha_{2,1}, \alpha_{3,1}, \ldots$
315
CALCULATE $\theta_1 = \alpha_{1,1} \odot \alpha_{2,1} \odot \alpha_{3,1} \ldots \odot \alpha_{n,1}$
320
i=1
325
CALCULATE
$\left\{ \{T_{SM}, I, 1, \alpha_{i,1}, \theta_1\} K_{SM}^{-1} \right\}_{Ki}$
330
DISTRIBUTE TO MEMBER i
335
i=i+1
340
i > n ?
NO
YES
345
RETURN

FIG.3

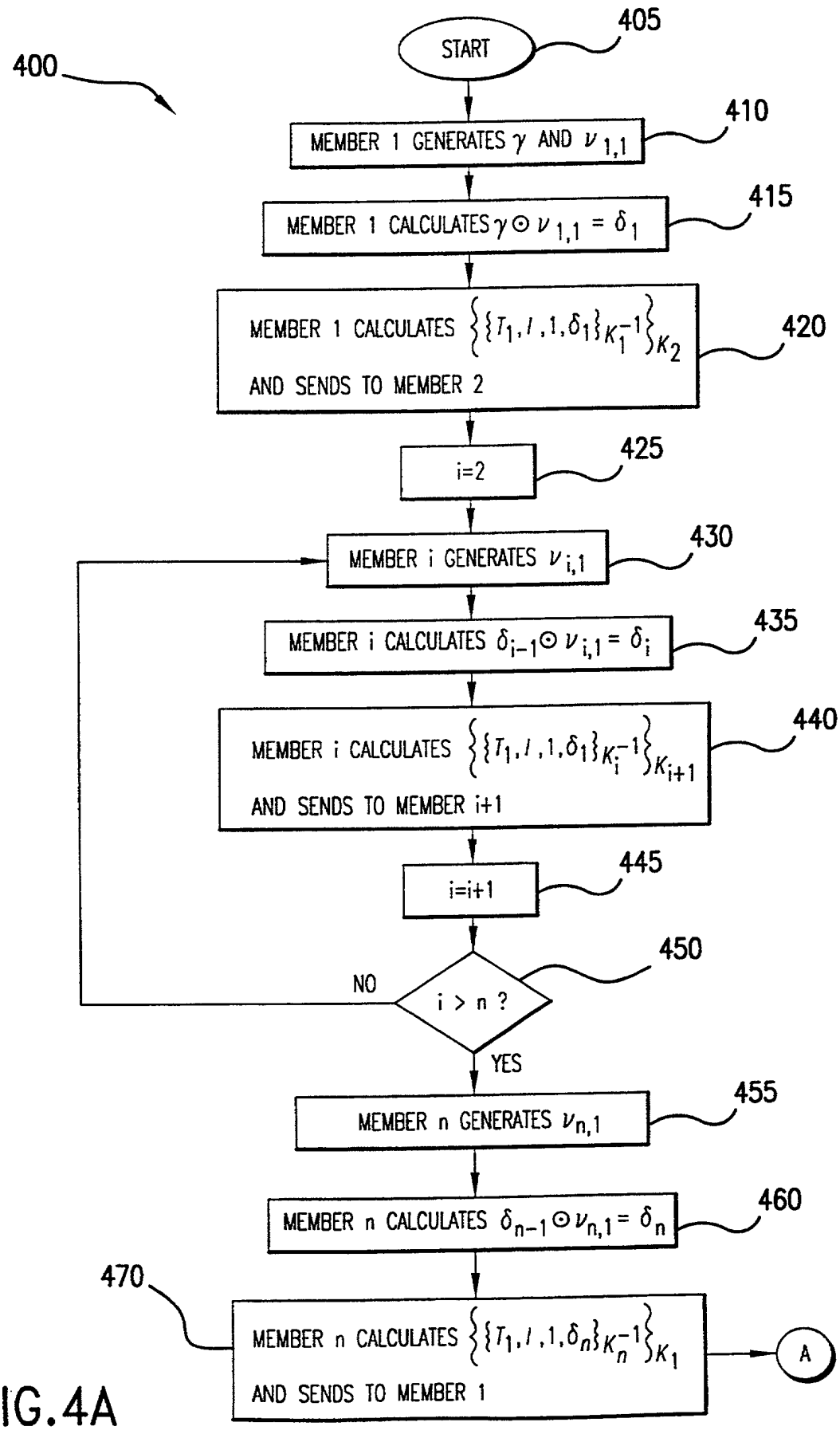
400
START
405
MEMBER 1 GENERATES γ AND $\nu_{1,1}$
410
MEMBER 1 CALCULATES $\gamma \odot \nu_{1,1} = \delta_1$
415
MEMBER 1 CALCULATES $\{\{T_1, I, 1, \delta_1\}_{K_1^{-1}}\}_{K_2}$ AND SENDS TO MEMBER 2
420
i=2
425
MEMBER i GENERATES $\nu_{i,1}$
430
MEMBER i CALCULATES $\delta_{i-1} \odot \nu_{i,1} = \delta_i$
435
MEMBER i CALCULATES $\{\{T_1, I, 1, \delta_1\}_{K_i^{-1}}\}_{K_{i+1}}$ AND SENDS TO MEMBER i+1
440
i=i+1
445
i > n ?
450
NO
YES
MEMBER n GENERATES $\nu_{n,1}$
455
MEMBER n CALCULATES $\delta_{n-1} \odot \nu_{n,1} = \delta_n$
460
470
MEMBER n CALCULATES $\{\{T_1, I, 1, \delta_n\}_{K_n^{-1}}\}_{K_1}$ AND SENDS TO MEMBER 1
A

FIG.4A

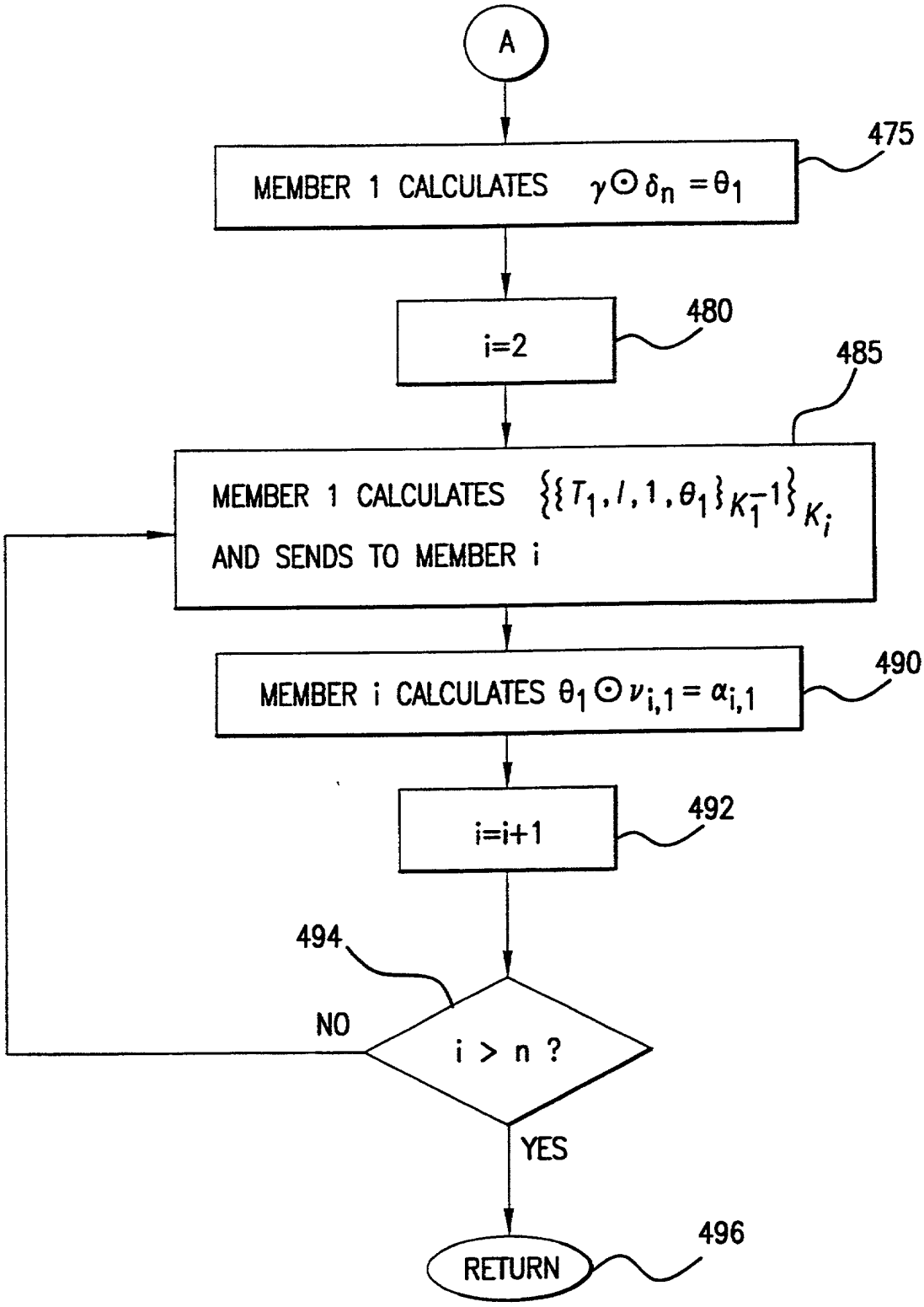
A
475
MEMBER 1 CALCULATES $\gamma \odot \delta_n = \theta_1$
480
i=2
485
MEMBER 1 CALCULATES $\{\{T_1, I, 1, \theta_1\}_{K_1^{-1}}\}_{K_i}$
AND SENDS TO MEMBER i
490
MEMBER i CALCULATES $\theta_1 \odot \nu_{i,1} = \alpha_{i,1}$
492
i=i+1
494
NO
i > n ?
YES
496
RETURN

FIG.4B

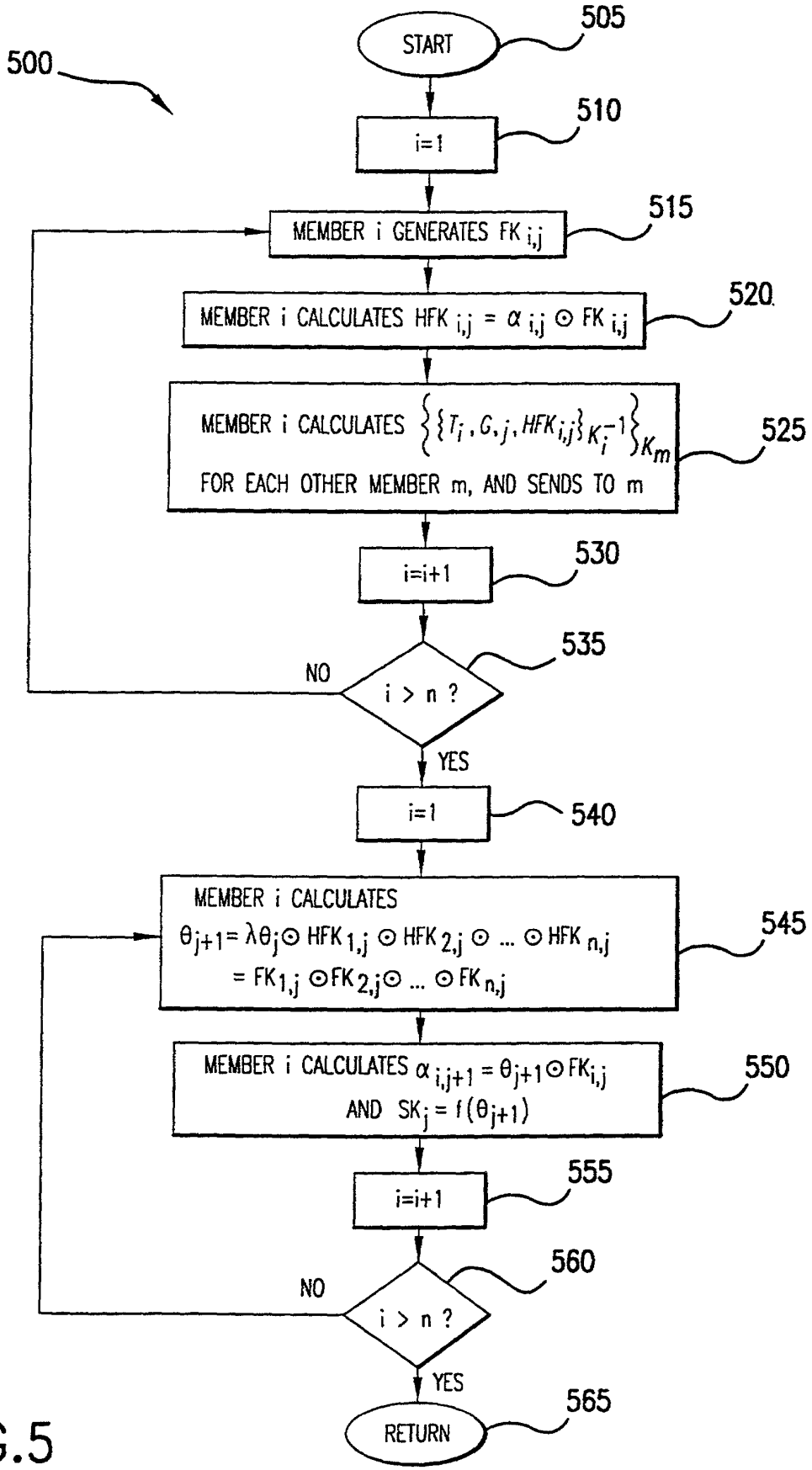
500
START
505
i=1
510
MEMBER i GENERATES FK i,j
515
MEMBER i CALCULATES HFK i,j = α i,j ⊙ FK i,j
520
MEMBER i CALCULATES $\{\{T_i, G_j, HFK_{i,j}\}K_i^{-1}\}K_m$
FOR EACH OTHER MEMBER m, AND SENDS TO m
525
i=i+1
530
NO
i > n ?
535
YES
i=1
540
MEMBER i CALCULATES
θj+1= λθj⊙ HFK 1,j ⊙ HFK 2,j ⊙ ... ⊙ HFK n,j
= FK1,j ⊙FK2,j⊙ ... ⊙ FK n,j
545
MEMBER i CALCULATES α i,j+1 = θj+1⊙FKi,j
AND SKj = f(θj+1)
550
i=i+1
555
NO
i > n ?
560
YES
RETURN
565

FIG.5

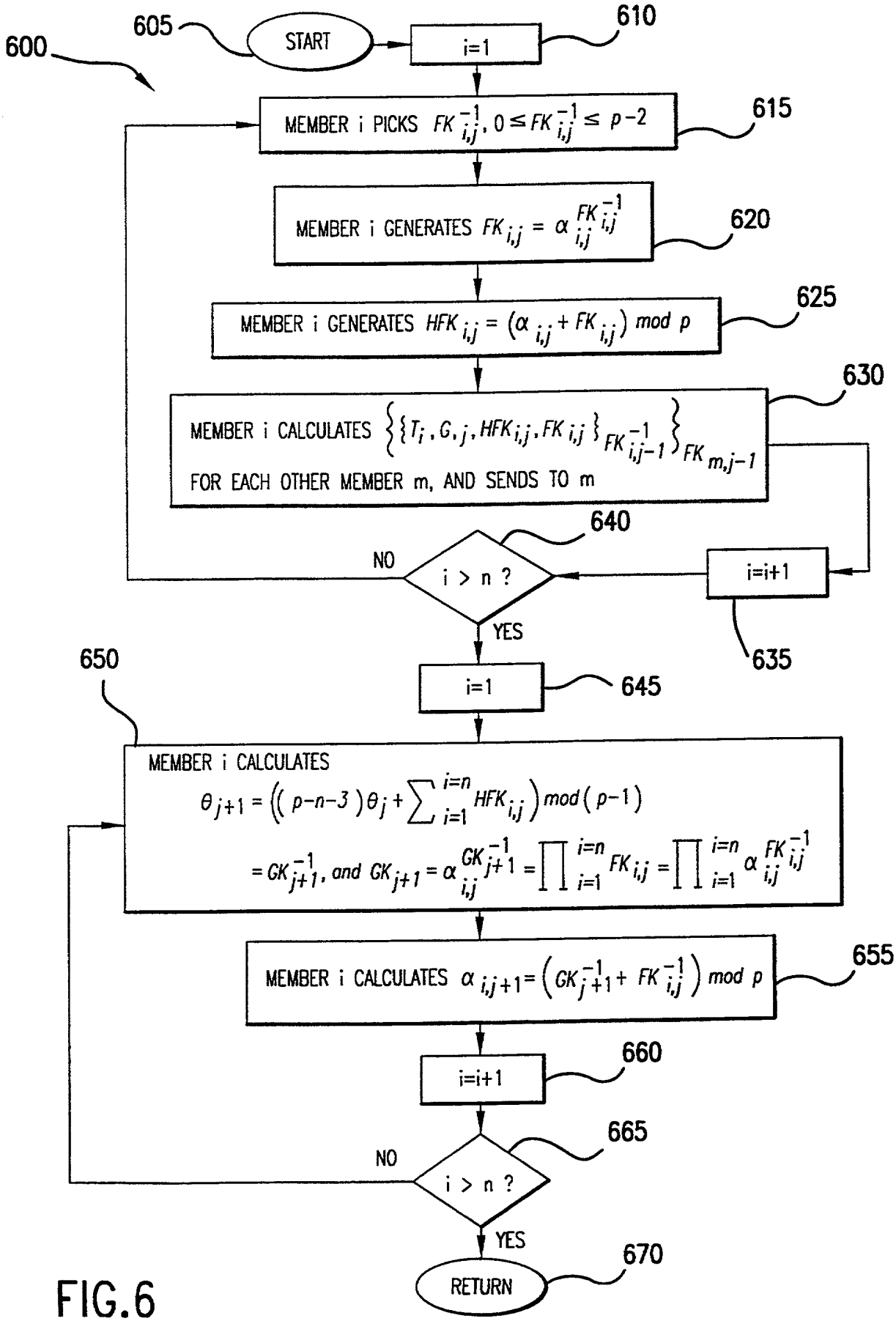
600
605
START
610
i=1
MEMBER i PICKS $FK_{i,j}^{-1}$, $0 \le FK_{i,j}^{-1} \le p-2$
615
MEMBER i GENERATES $FK_{i,j} = \alpha_{i,j}^{FK_{i,j}^{-1}}$
620
MEMBER i GENERATES $HFK_{i,j} = (\alpha_{i,j} + FK_{i,j}) \bmod p$
625
630
MEMBER i CALCULATES $\left\{\left\{T_i, G, j, HFK_{i,j}, FK_{i,j}\right\}_{FK_{i,j-1}^{-1}}\right\}_{FK_{m,j-1}}$
FOR EACH OTHER MEMBER m, AND SENDS TO m
640
NO
i > n ?
i=i+1
635
YES
650
i=1
645
MEMBER i CALCULATES
$\theta_{j+1} = \left(\left(p-n-3\right)\theta_j + \sum_{i=1}^{i=n} HFK_{i,j}\right) mod\left(p-1\right)$
$= GK_{j+1}^{-1}$, and $GK_{j+1} = \alpha_{i,j}^{GK_{j+1}^{-1}} = \prod_{i=1}^{i=n} FK_{i,j} = \prod_{i=1}^{i=n} \alpha_{i,j}^{FK_{i,j}^{-1}}$
MEMBER i CALCULATES $\alpha_{i,j+1} = \left(GK_{j+1}^{-1} + FK_{i,j}^{-1}\right) \bmod p$
655
i=i+1
660
665
NO
i > n ?
YES
670
RETURN

FIG.6

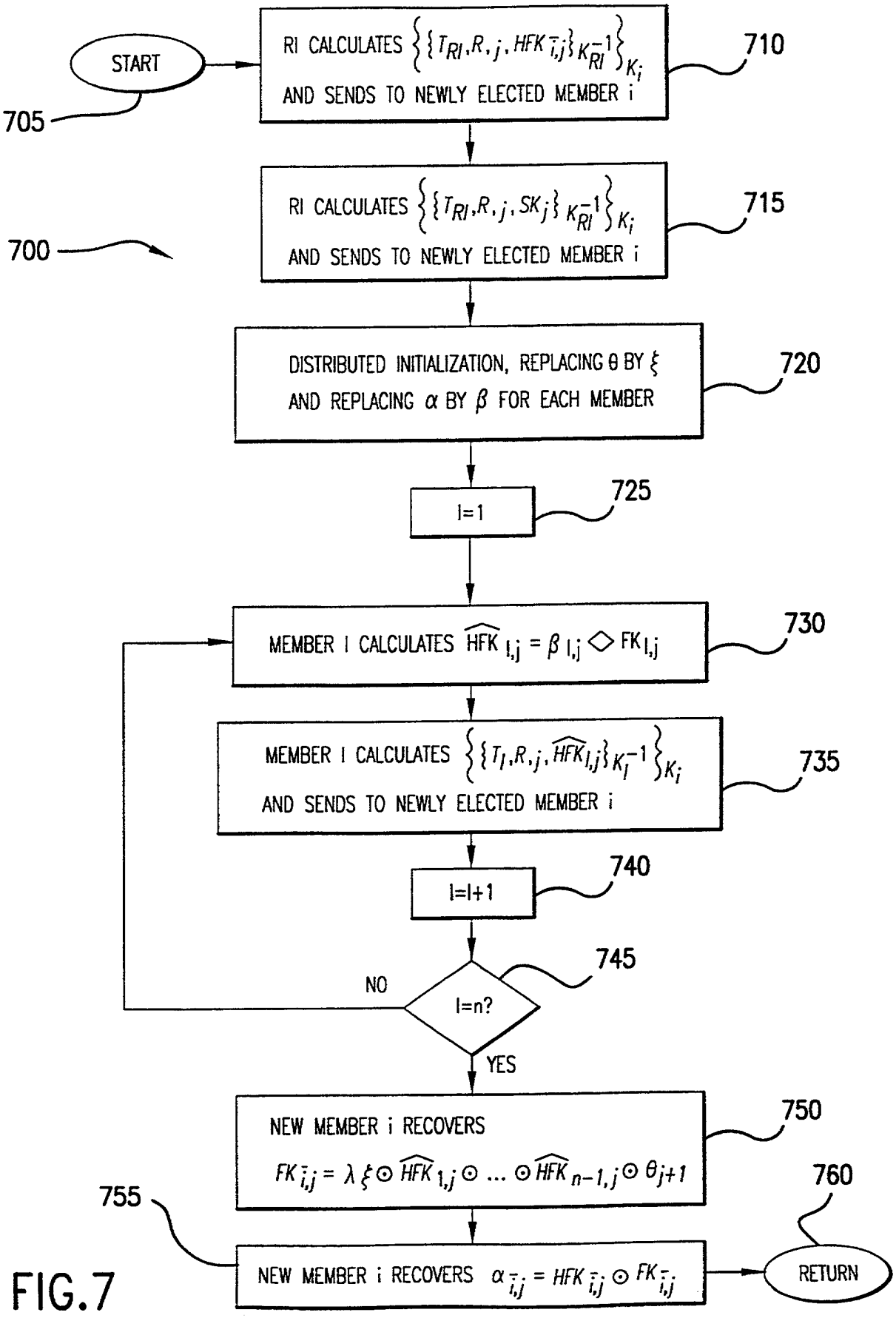
START
705
RI CALCULATES $\{\{T_{RI}, R, j, HFK_{i,j}\} K_{RI}^{-1}\}_{K_i}$
AND SENDS TO NEWLY ELECTED MEMBER i
710
RI CALCULATES $\{\{T_{RI}, R, j, SK_j\} K_{RI}^{-1}\}_{K_i}$
AND SENDS TO NEWLY ELECTED MEMBER i
715
700
DISTRIBUTED INITIALIZATION, REPLACING θ BY ξ
AND REPLACING α BY β FOR EACH MEMBER
720
l=1
725
MEMBER I CALCULATES $\widehat{HFK}_{I,j} = \beta_{I,j} \diamond FK_{I,j}$
730
MEMBER I CALCULATES $\{\{T_I, R, j, \widehat{HFK}_{I,j}\} K_I^{-1}\}_{K_i}$
AND SENDS TO NEWLY ELECTED MEMBER i
735
l=l+1
740
NO
l=n?
745
YES
NEW MEMBER i RECOVERS
$FK_{i,j} = \lambda \xi \odot \widehat{HFK}_{1,j} \odot \ldots \odot \widehat{HFK}_{n-1,j} \odot \theta_{j+1}$
750
755
NEW MEMBER i RECOVERS $\alpha_{i,j} = HFK_{i,j} \odot FK_{i,j}$
760
RETURN

FIG.7

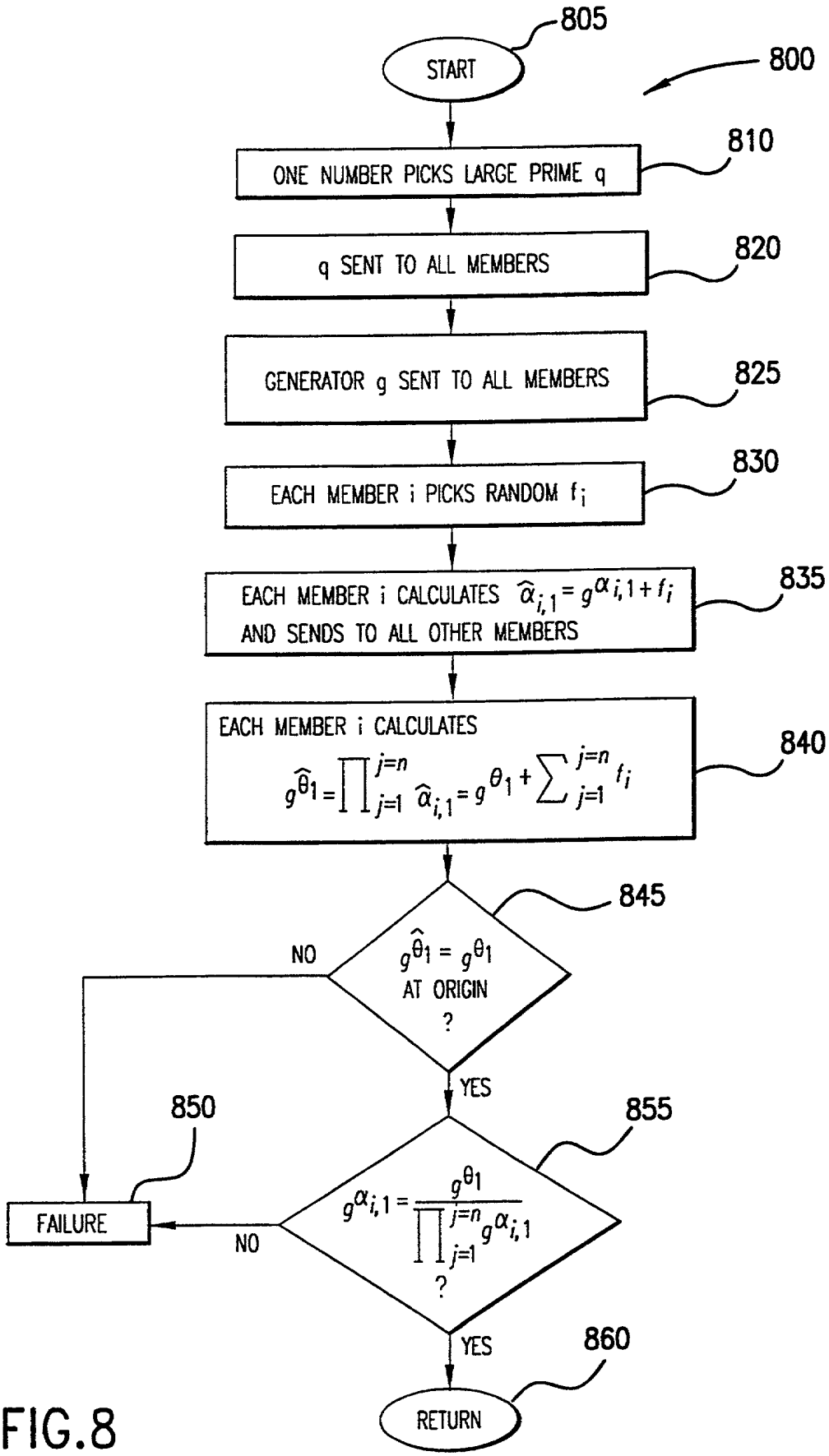
805
START
800
ONE NUMBER PICKS LARGE PRIME q
810
q SENT TO ALL MEMBERS
820
GENERATOR g SENT TO ALL MEMBERS
825
EACH MEMBER i PICKS RANDOM f_i
830
EACH MEMBER i CALCULATES $\hat{\alpha}_{i,1} = g^{\alpha i,1} + f_i$
AND SENDS TO ALL OTHER MEMBERS
835
EACH MEMBER i CALCULATES
$g^{\hat{\theta}_1} = \prod_{j=1}^{j=n} \hat{\alpha}_{i,1} = g^{\theta_1} + \sum_{j=1}^{j=n} f_i$
840
845
$g^{\hat{\theta}_1} = g^{\theta_1}$ AT ORIGIN ?
NO
YES
850
FAILURE
855
$g^{\alpha_{i,1}} = \frac{g^{\theta_1}}{\prod_{j=1}^{j=n} g^{\alpha_{i,1}}}$?
NO
YES
860
RETURN

FIG.8

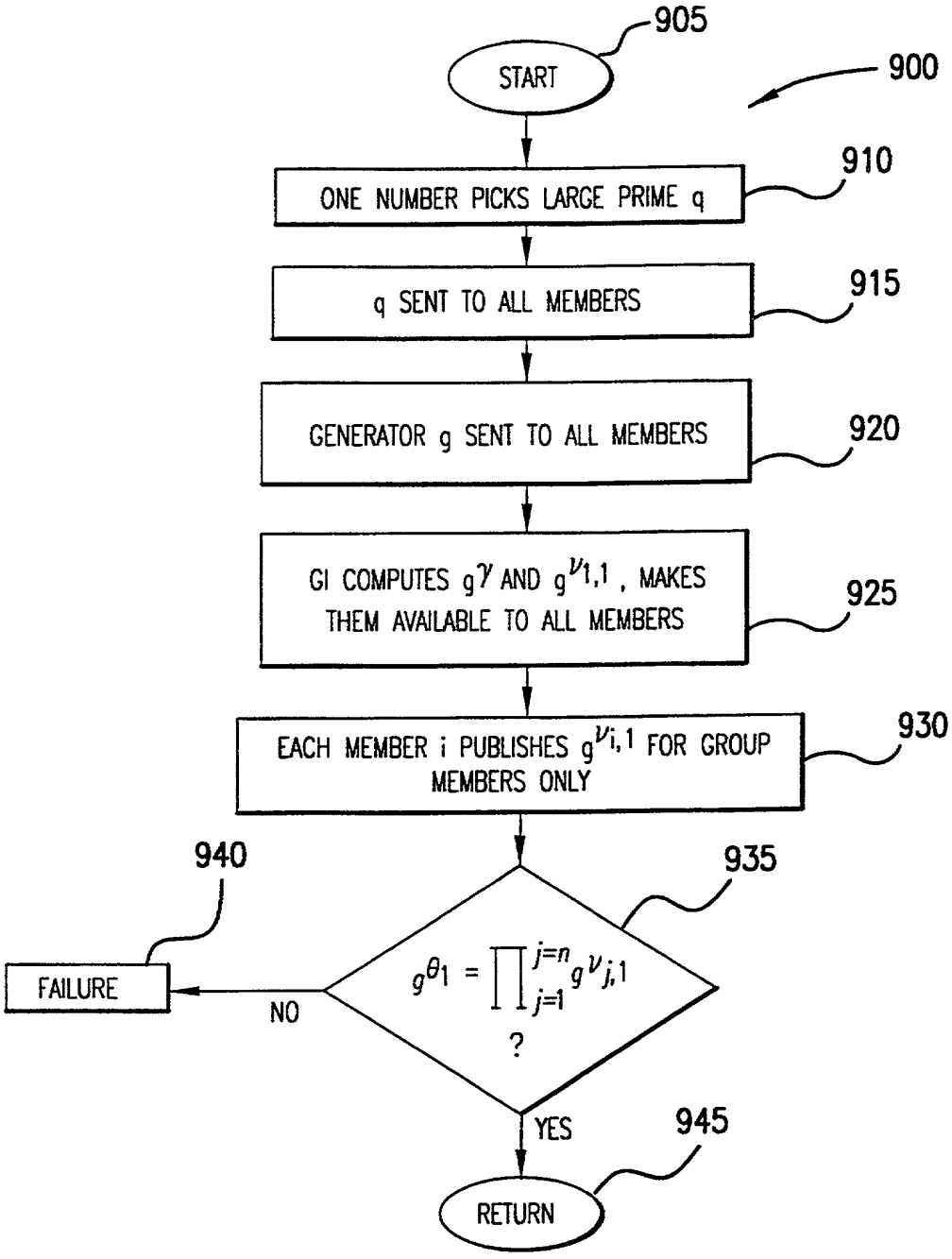
905
START
900
ONE NUMBER PICKS LARGE PRIME q
910
q SENT TO ALL MEMBERS
915
GENERATOR g SENT TO ALL MEMBERS
920
GI COMPUTES g^{γ} AND $g^{\nu_{1,1}}$, MAKES THEM AVAILABLE TO ALL MEMBERS
925
EACH MEMBER i PUBLISHES $g^{\nu_{i,1}}$ FOR GROUP MEMBERS ONLY
930
940
935
FAILURE
NO
$g^{\theta_1} = \prod_{j=1}^{j=n} g^{\nu_{j,1}}$?
YES
945
RETURN

FIG.9

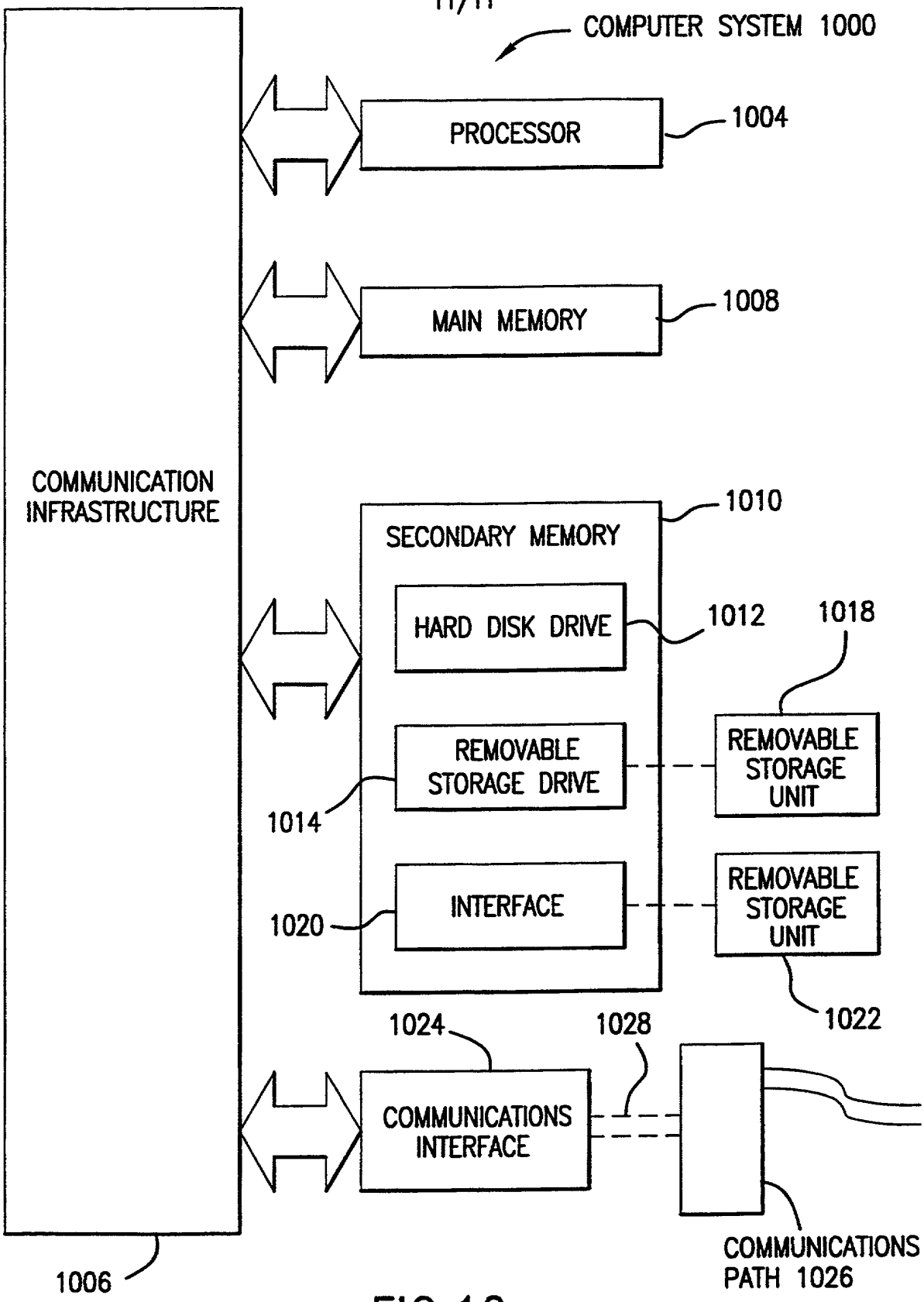
COMPUTER SYSTEM 1000
PROCESSOR
1004
MAIN MEMORY
1008
COMMUNICATION INFRASTRUCTURE
SECONDARY MEMORY
1010
HARD DISK DRIVE
1012
1018
REMOVABLE STORAGE DRIVE
1014
REMOVABLE STORAGE UNIT
INTERFACE
1020
REMOVABLE STORAGE UNIT
1022
1024
1028
COMMUNICATIONS INTERFACE
COMMUNICATIONS PATH 1026
1006

FIG.10

RECEIVED MAY 16 2001

Declaration for Patent Application

Docket Number: 1797.014PC02

As a below named inventor, I hereby declare that:

My residence, mailing address and citizenship are as stated below next to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter that is claimed and for which a patent is sought on the invention entitled Distributed Shared Key Generation and Management Using Fractional Keys, the specification of which is attached hereto unless the following box is checked:

☒ was filed on October 1, 1999;
as United States Application Number or PCT International Application Number PCT/US99/22710; and
was amended on _____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information that is material to patentability as defined in 37 C.F.R. § 1.56.

I hereby claim foreign priority benefits under 35 U.S.C. § 119(a)-(d) or § 365(b) of any foreign application(s) for patent or inventor's certificate, or § 365(a) of any PCT international application, which designated at least one country other than the United States listed below, and have also identified below any foreign application for patent or inventor's certificate, or PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)			Priority Claimed
_____	_____	_____	☐ Yes ☐ No
(Application No.)	(Country)	(Day/Month/Year Filed)	
_____	_____	_____	☐ Yes ☐ No
(Application No.)	(Country)	(Day/Month/Year Filed)	

I hereby claim the benefit under 35 U.S.C. § 119(e) of any United States provisional application(s) listed below.

60/102,633	October 1, 1998
(Application No.)	(Filing Date)
60/131,833	April 29, 1999
(Application No.)	(Filing Date)

I hereby claim the benefit under 35 U.S.C. § 120 of any United States application(s), or under § 365(c) of any PCT international application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT international application in the manner provided by the first paragraph of 35 U.S.C. § 112, I acknowledge the duty to disclose information that is material to patentability as defined in 37 C.F.R. § 1.56 that became available between the filing date of the prior application and the national or PCT international filing date of this application.

PCT/US99/22710	October 1, 1999	Pending
(Application No.)	(Filing Date)	(Status - patented, pending, abandoned)
_____	_____	_____
(Application No.)	(Filing Date)	(Status - patented, pending, abandoned)

09606548 - 052501

Send Correspondence to:

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 New York Avenue, N.W.
Suite 600
Washington, D.C. 20005-3934

Direct Telephone Calls to:

(202) 371-2600

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of sole or first inventor	Raadhakrishnan POOVENDRAN
Signature of sole or first inventor	Date
Residence	Greenbelt, Maryland
Citizenship	United States
Mailing Address	P O. Box 474, Greenbelt, Maryland 20768
Full name of second inventor	Matthew Scott CORSON
Signature of second inventor	Date MAY 14, 2001
Residence	Kensington, Maryland MD
Citizenship	United States
Mailing Address	10122 Ashwood Drive, Kensington, Maryland 20895
Full name of third inventor	John S. BARAS
Signature of third inventor	Date
Residence	Potomac, Maryland
Citizenship	United States
Mailing Address	10912 Burbank Drive, Potomac, Maryland 20854
SKGF Rev. 11/8/00 mac	(Supply similar information and signature for subsequent joint inventors, if any)

Appl. No. PCT/US99/22710
Docket No. 1797.014PC02

Send Correspondence to:

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 New York Avenue, N.W.
Suite 600
Washington, D.C. 20005-3934

Direct Telephone Calls to:

(202) 371-2600

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

1-00

Full name of sole or first inventor	Raamhakrishnan POUVENDRAN
Signature of sole or first inventor	Date 3/30/2001
Residence	~~Greenbelt, Maryland~~ SEATTLE, WA WA
Citizenship	United States
Mailing Address	EE DEPT. BOX 352500, UNIVERSITY OF WASHINGTON ~~P.O. Box 686, Greenbelt, Maryland 20768~~ SEATTLE, WA 98195-2500
Full name of second inventor	Matthew Scott CORSON
Signature of second inventor	Date
Residence	Kensington, Maryland
Citizenship	United States
Mailing Address	10122 Ashwood Drive, Kensington, Maryland 20895
Full name of third inventor	John S. BARAS
Signature of third inventor	Date
Residence	Potomac, Maryland
Citizenship	United States
Mailing Address	10912 Burbank Drive, Potomac, Maryland 20854
	(Supply similar information and signature for subsequent joint inventors, if any)

TOTAL P.03

Appl. No. PCT/US99/22710
Docket No. 1797.014PC02

Send Correspondence to:

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 New York Avenue, N.W.
Suite 600
Washington, D.C. 20005-3934

Direct Telephone Calls to:

(202) 371-2600

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of sole or first inventor	Raadhakrishnan POOVENDRAN
Signature of sole or first inventor	Date
Residence	Greenbelt, Maryland
Citizenship	United States
Mailing Address	P.O. Box 474, Greenbelt, Maryland 20768
Full name of second inventor	Matthew Scott CORSON
Signature of second inventor	Date
Residence	Kensington, Maryland
Citizenship	United States
Mailing Address	10122 Ashwood Drive, Kensington, Maryland 20895
Full name of third inventor	John S. BARAS
Signature of third inventor	April 25, 2001 Date
Residence	Potomac, Maryland MD
Citizenship	United States
Mailing Address	10912 Burbank Drive, Potomac, Maryland 20854
	(Supply similar information and signature for subsequent joint inventors, if any)

3-00

POWER OF ATTORNEY FROM ASSIGNEE

University of Maryland, College Park, a university of Maryland, having a principal place of business at Office of Technology Commercialization, University of Maryland, College Park, 6200 Baltimore Avenue, Suite 300, College Park, MD 20742-9520, is assignee of the entire right, title and interest for the United States of America (as defined in 35 U.S.C. § 100), by reason of an Assignment to the Assignee executed on 4/26/00, 4/20/00, and 4/24/00 of an invention known as Distributed Shared Key Generation and Management Using Fractional Keys (Attorney Docket No. 1797.014PC02), which is disclosed and claimed in a patent application of the same title by the inventor(s) Poovendran *et al.* (said application filed on October 1, 1999 at the U.S. Patent and Trademark Office, having Application Number PCT/US99/22710).

The Assignee hereby appoints the following U.S. attorneys to prosecute this application and any continuation, divisional, continuation-in-part, or reissue application thereof, and to transact all business in the U.S. Patent and Trademark Office connected therewith: Robert Greene Sterne, Esq., Reg. No. 28,912; Edward J. Kessler, Esq., Reg. No. 25,688; Jorge A. Goldstein, Esq., Reg. No. 29,021; David K.S. Cornwell, Esq., Reg. No. 31,944; Robert W. Esmond, Esq., Reg. No. 32,893; Tracy-Gene G. Durkin, Esq., Reg. No. 32,831; Michele A. Cimbala, Esq., Reg. No. 33,851; Michael B. Ray, Esq., Reg. No. 33,997; Robert E. Sokohl, Esq., Reg. No. 36,013; Eric K. Steffe, Esq., Reg. No. 36,688, Michael Q. Lee, Esq., Reg. No. 35,239; Steven R. Ludwig, Esq., Reg. No. 36,203; John M. Covert, Esq., Reg. No. 38,759; and Linda E. Alcorn, Esq., Reg. No. 39,588. The Assignee hereby grants said attorneys the power to insert on this Power of Attorney any further identification that may be necessary or desirable in order to comply with the rules of the U.S. Patent and Trademark Office.

(15)

Send correspondence to:

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 New York Avenue, N.W.
Suite 600
Washington, D.C. 20005-3934
U.S.A.

Direct phone calls to 202-371-2600.

FOR: University of Maryland, College Park

SIGNATURE: [signature]

BY: James A. Poulos, III

TITLE: Executive Director

DATE: May 24, 2001

©2001 STERNE, KESSLER, GOLDSTEIN & FOX P L.L.C.
SKGF Rev. 8
1/26/01mac

Certificate Under 37 C.F.R. § 3.73(b)

Applicant/Patent Owner: Poovendran *et al.*

Application No./Patent No.: PCT/US99/22710 Filed/Issue Date: October 1, 1999

Entitled: Distributed Shared Key Generation and Management Using Fractional Keys

University of Maryland, College Park, a university
(Name of Assignee) (Type of Assignee, e g , corporation, partnership, university, government agency,etc)

states that it is:

1. [X] the assignee of the entire right, title, and interest, or

2. [ ] an assignee of an undivided part interest

in the patent application/patent identified above by virtue of either:

A. [X] An Assignment from the inventor(s) of the patent application/patent identified above. The assignment was recorded in the Patent and Trademark Office at Reel _____, Frame ____, or for which a copy thereof is attached.

OR

B. [ ] A chain of title from the inventor(s) of the patent application/patent identified above to the current assignee as shown below:

1. From: _____ To: _____
The document was recorded in the Patent and Trademark Office at
Reel _____, Frame _____, or for which a copy thereof is attached.

2. From: _____ To: _____
The document was recorded in the Patent and Trademark Office at
Reel _____, Frame _____, or for which a copy thereof is attached.

3. From: _____ To: _____
The document was recorded in the Patent and Trademark Office at
Reel _____, Frame _____, or for which a copy thereof is attached.

[ ] Additional documents in the chain of title are listed on a supplemental sheet.

[X] Copies of assignments or other documents in the chain of title are attached.

[**NOTE:** A separate copy (*i.e.*, the original assignment document or a true copy of the original document) must be submitted to Assignment Division in accordance with 37 CFR Part 3, if the assignment is to be recorded in the records of the PTO. See MPEP 302-302.8]

The undersigned (whose title is supplied below) is empowered to act on behalf of the assignee.

Date: May 29, 2001

Name: James A. Poulos, III

Title: Executive Director

Signature: _____

P:\USERS\DKELSEY\Attny Documents\EWY\1797\014pc02\373bcert SKGF Rev 3/31/00 mac

Page 2 of 2 Doc 3

DO NOT FORWARD
TO ASSIGNMENT BRANCH
NOT FOR RECORDATION

INTELLECTUAL PROPERTY ASSIGNMENT AGREEMENT

THIS AGREEMENT by and between **John S. Baras**, an individual having a principal residence at **109112 Burbank Drive, Potomac, Maryland 20854** (hereinafter referred to as "Assignor"), and the University of Maryland, having a principal office at Office of Technology Liaison, College Park, Maryland 20742 (hereinafter referred to as "Assignee").

WITNESSETH:

WHEREAS, Assignor has created and developed certain inventions, improvements, discoveries, software, or other intellectual property, as described in **Assignee Invention Disclosure No. IS-98-012 and IS-99-027 and in International Application No. PCT/US99/22710 titled "Distributed Shared Key Generation and Management Using Fractional Keys" filed October 1, 1999** (hereinafter collectively referred to as the "Works"); and

WHEREAS, Assignor agrees that, to the extent the Works are, by operation of law or otherwise, not deemed to be works made for hire within the meaning of the Copyright Act, (Title 17, U.S.C. Section 101, et seq.), Assignor agrees to assign all of his/her right, title and interest in and to the Works to Assignee, and further agrees to take such further actions and to execute such further instruments that Assignee might find reasonable or necessary to perfect or to evidence more clearly its right and claim to exclusive ownership of all of Assignor's worldwide intellectual property interests respecting the Works; and

WHEREAS, Assignor and Assignee now wish to perfect and to evidence more clearly the right and claim of Assignee to exclusive ownership of all of Assignor's intellectual property interests respecting the Works.

NOW, THEREFORE, in consideration of the rights granted to Assignor under the University of Maryland Patent Policy and Copyright Policy, as approved by the Board of Regents of the University of Maryland, and as amended by them from time to time, and other good and valuable consideration furnished by Assignee to Assignor, the receipt and sufficiency of which are hereby acknowledged, Assignor and Assignee, intending to be legally bound, do hereby covenant and agree as follows:

Section 1. Assignment of the Works.

Assignor hereby assigns, transfers and conveys to Assignee, its successors, assigns or other legal representatives, without the necessity of any consideration in addition to that recited herein, all of Assignor's right, title and interest in and to the Works. This assignment shall be operative with respect to all intellectual property rights in and to the Works, including (without limitation), (i) all copyrights in the United States and elsewhere, including all rights of registration, publication, renewal, rights to create derivative works and all other rights incident to copyright ownership, for the

residue now unexpired of the present term of any and all such copyrights and any term thereafter granted during which the Works are entitled to copyright; (ii) all trade secrets, inventions, know-how, ideas and confidential information embodied or reflected in the Works, including any shop rights, for the longest period of protection accorded to such interest under applicable law; and (iii) all patent rights in the United States and elsewhere, including all rights of registration, publication, renewal, and all other works incident to copyright ownership, for the longest period of protection accorded to such interests under applicable law.

Section 2. University of Maryland Copyright and Patent Policies.

The assignment of rights perfected hereunder shall be governed by the University of Maryland Patent Policy and Copyright Policy as approved by the Board of Regents of the University of Maryland, and as amended by them from time to time. Royalty income shall be allocated as set forth in those policies.

Section 3. Warranty.

Assignor warrants and covenants that he/she is an author or inventor of the Works and that as of the date of this Assignment, has taken no action respecting the Works which purports or attempts to transfer or encumber any right, title or interest in or to the Works to any other party; and covenants not to take such action in the future.

Section 4. Jurisdiction.

The validity, interpretation, and effect of this agreement shall be governed by the laws of the State of Maryland and of the United States of America. Any legal proceedings involving claims or disputes regarding this agreement shall be brought in the appropriate court in the State of Maryland.

WHEREAS, the parties have caused this Assignment to be executed on the dates below.

ASSIGNOR

Agreed to by: _____ Date: 4/20/2000

Printed Name: John S. Baras

ASSIGNEE (The University of Maryland)

Acknowledged and Agreed to by:

_____ Date: May 1, 200

James A. Poulos, III

Title: Acting Executive Director

PF-0000-0000-0000-0000

DO NOT FORWARD
TO ASSIGNMENT BRANCH
NOT FOR RECORDATION

RECEIVED APR 2 5 2000

INTELLECTUAL PROPERTY ASSIGNMENT AGREEMENT

THIS AGREEMENT by and between **Mathew Scott Corson**, an individual having a principal residence at **10122 Ashwood Drive, Kensington, Maryland 20895** (hereinafter referred to as "Assignor"), and the University of Maryland, having a principal office at Office of Technology Liaison, College Park, Maryland 20742 (hereinafter referred to as "Assignee").

WITNESSETH:

WHEREAS, Assignor has created and developed certain inventions, improvements, discoveries, software, or other intellectual property, as described in **Assignee Invention Disclosure No. IS-98-012 and IS-99-027 and in International Application No. PCT/US99/22710 titled "Distributed Shared Key Generation and Management Using Fractional Keys" filed October 1, 1999** (hereinafter collectively referred to as the "Works"); and

WHEREAS, Assignor agrees that, to the extent the Works are, by operation of law or otherwise, not deemed to be works made for hire within the meaning of the Copyright Act, (Title 17, U.S.C. Section 101, et seq.), Assignor agrees to assign all of his/her right, title and interest in and to the Works to Assignee, and further agrees to take such further actions and to execute such further instruments that Assignee might find reasonable or necessary to perfect or to evidence more clearly its right and claim to exclusive ownership of all of Assignor's worldwide intellectual property interests respecting the Works; and

WHEREAS, Assignor and Assignee now wish to perfect and to evidence more clearly the right and claim of Assignee to exclusive ownership of all of Assignor's intellectual property interests respecting the Works.

NOW, THEREFORE, in consideration of the rights granted to Assignor under the University of Maryland Patent Policy and Copyright Policy, as approved by the Board of Regents of the University of Maryland, and as amended by them from time to time, and other good and valuable consideration furnished by Assignee to Assignor, the receipt and sufficiency of which are hereby acknowledged, Assignor and Assignee, intending to be legally bound, do hereby covenant and agree as follows:

Section 1. Assignment of the Works.

Assignor hereby assigns, transfers and conveys to Assignee, its successors, assigns or other legal representatives, without the necessity of any consideration in addition to that recited herein, all of Assignor's right, title and interest in and to the Works. This assignment shall be operative with respect to all intellectual property rights in and to the Works, including (without limitation), (i) all copyrights in the United States and elsewhere, including all rights of registration, publication,

renewal, rights to create derivative works and all other rights incident to copyright ownership, for the residue now unexpired of the present term of any and all such copyrights and any term thereafter granted during which the Works are entitled to copyright; (ii) all trade secrets, inventions, know-how, ideas and confidential information embodied or reflected in the Works, including any shop rights, for the longest period of protection accorded to such interest under applicable law; and (iii) all patent rights in the United States and elsewhere, including all rights of registration, publication, renewal, and all other works incident to copyright ownership, for the longest period of protection accorded to such interests under applicable law.

Section 2. University of Maryland Copyright and Patent Policies.

The assignment of rights perfected hereunder shall be governed by the University of Maryland Patent Policy and Copyright Policy as approved by the Board of Regents of the University of Maryland, and as amended by them from time to time. Royalty income shall be allocated as set forth in those policies.

Section 3. Warranty.

Assignor warrants and covenants that he/she is an author or inventor of the Works and that as of the date of this Assignment, has taken no action respecting the Works which purports or attempts to transfer or encumber any right, title or interest in or to the Works to any other party; and covenants not to take such action in the future.

Section 4. Jurisdiction.

The validity, interpretation, and effect of this agreement shall be governed by the laws of the State of Maryland and of the United States of America. Any legal proceedings involving claims or disputes regarding this agreement shall be brought in the appropriate court in the State of Maryland.

WHEREAS, the parties have caused this Assignment to be executed on the dates below.

ASSIGNOR

Agreed to by: _____ Date: 4/24/20

Printed Name: Mathew Scott Corson

ASSIGNEE (The University of Maryland)

Acknowledged and Agreed to by:

_____ Date: May 1, 2020

James A. Poulos, III

Title: Acting Executive Director

FOR INTERNAL USE ONLY

DO NOT FORWARD
TO ASSIGNMENT BRANCH
NOT FOR RECORDATION

INTELLECTUAL PROPERTY ASSIGNMENT AGREEMENT

THIS AGREEMENT by and between **Raadhakrishnan Poovendran,** an individual having a principal residence at **P.O. Box 474, Greenbelt, Maryland 20768** (hereinafter referred to as "Assignor"), and the University of Maryland, having a principal office at Office of Technology Liaison, College Park, Maryland 20742 (hereinafter referred to as "Assignee").

WITNESSETH:

WHEREAS, Assignor has created and developed certain inventions, improvements, discoveries, software, or other intellectual property, as described in **Assignee Invention Disclosure No. IS-98-012 and IS-99-027 and in International Application No. PCT/US99/22710 titled "Distributed Shared Key Generation and Management Using Fractional Keys" filed October 1, 1999** (hereinafter collectively referred to as the "Works"); and

WHEREAS, Assignor agrees that, to the extent the Works are, by operation of law or otherwise, not deemed to be works made for hire within the meaning of the Copyright Act, (Title 17, U.S.C. Section 101, et seq.), Assignor agrees to assign all of his/her right, title and interest in and to the Works to Assignee, and further agrees to take such further actions and to execute such further instruments that Assignee might find reasonable or necessary to perfect or to evidence more clearly its right and claim to exclusive ownership of all of Assignor's worldwide intellectual property interests respecting the Works; and

WHEREAS, Assignor and Assignee now wish to perfect and to evidence more clearly the right and claim of Assignee to exclusive ownership of all of Assignor's intellectual property interests respecting the Works.

NOW, THEREFORE, in consideration of the rights granted to Assignor under the University of Maryland Patent Policy and Copyright Policy, as approved by the Board of Regents of the University of Maryland, and as amended by them from time to time, and other good and valuable consideration furnished by Assignee to Assignor, the receipt and sufficiency of which are hereby acknowledged, Assignor and Assignee, intending to be legally bound, do hereby covenant and agree as follows:

Section 1. Assignment of the Works.

Assignor hereby assigns, transfers and conveys to Assignee, its successors, assigns or other legal representatives, without the necessity of any consideration in addition to that recited herein, all of Assignor's right, title and interest in and to the Works. This assignment shall be operative with respect to all intellectual property rights in and to the Works, including ~~(without limitation)~~, (i) ~~all copyrights in the United States and elsewhere,~~ including all rights of registration, ~~publication~~, renewal, ~~rights to create derivative works and all other rights incident to copyright ownership,~~ for the

~~residue now unexpired of the present term of any and all such copyrights and any term thereafter granted during which the Works are entitled to copyright~~; (ii) all trade secrets, inventions, know-how, ideas and confidential information embodied or reflected in the Works, including any shop rights, for the longest period of protection accorded to such interest under applicable law; and (iii) all patent rights in the United States and elsewhere, including all rights of registration, ~~publication~~, renewal, ~~and all other works incident to copyright ownership~~, for the longest period of protection accorded to such interests under applicable law.

Section 2. **University of Maryland Copyright and Patent Policies.**

The assignment of rights perfected hereunder shall be governed by the University of Maryland Patent Policy and Copyright Policy as approved by the Board of Regents of the University of Maryland, and as amended by them from time to time. Royalty income shall be allocated as set forth in those policies.

Section 3. **Warranty.**

Assignor warrants and covenants that he/she is an author or inventor of the Works and that as of the date of this Assignment, has taken no action respecting the Works which purports or attempts to transfer or encumber any right, title or interest in or to the Works to any other party; and covenants not to take such action in the future.

Section 4. **Jurisdiction.**

The validity, interpretation, and effect of this agreement shall be governed by the laws of the State of Maryland and of the United States of America. Any legal proceedings involving claims or disputes regarding this agreement shall be brought in the appropriate court in the State of Maryland.

WHEREAS, the parties have caused this Assignment to be executed on the dates below.

ASSIGNOR

Agreed to by: P. Raadhakrishnan Date: 4/26/00

Printed Name: Raadhakrishnan Poovendran

ASSIGNEE (The University of Maryland)

Acknowledged and Agreed to by:

James A. Poulos, III Date: May 1, 2000

Title: Acting Executive Director

09809369.070200